UCID-19392, Rev. 1

TAURUS: An Interactive Post-Processor For The Analysis Codes NIKE3D, DYNA3D, TACO3D, and GEMINI

B. E. Brown
J. O. Hallquist

July, 1982
Rev. 1, May, 1984

Lawrence Livermore National Laboratory

Work performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.

DISCLAIMER

Printed in the United States of America
Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161
Price: Printed Copy $; Microfiche $3.50

Page Range	Domestic Price	Page Range	Domestic Price
001-025	$ 5.00	326-350	$ 18.00
026-050	6.00	351-375	19.00
051-075	7.00	376-400	20.00
076-100	8.00	401-425	21.00
101-125	9.00	426-450	22.00
126-150	10.00	451-475	23.00
151-175	11.00	476-500	24.00
176-200	12.00	501-525	25.00
201-225	13.00	526-550	26.00
226-250	14.00	551-525	27.00
251-275	15.00	526-550	28.00
276-300	16.00	601-up [1]	
301-325	17.00		

[1] Add 2.00 for each additional 25 page increment from 601 pages up.

TAURUS: An Interactive Post-Processor for The Analysis Codes NIKE3D, DYNA3D, TACO3D, and GEMINI

Bruce E. Brown†

J. O. Hallquist*

Consultant†
561 E. Quail Road
Orem, Utah 84057

Methods Development Group*
Mechanical Engineering Department
Lawrence Livermore National Laboratory

July, 1982

Rev. 1, May, 1984

CONTENTS

TAURUS: An interactive post-processor for the analysis codes NIKE3D, DYNA3D, TACO3D, and GEMINI

ABSTRACT

This report provides a user's manual for the post-processor, TAURUS. TAURUS reads the binary plot files generated by the two and three dimensional finite element codes currently used at LLNL and plots contours, time histories, and deformed shapes. Contours of a large number of quantities may be plotted on meshes consisting of plate, shell, and solid type elements. TAURUS can compute a variety of strain measures, reaction forces along constrained boundaries, and momentum. TAURUS is operational on the CRAY-1, 7600, and VAX computers.

INTRODUCTION

BACKGROUND

TAURUS replaces the non-interactive post-processor DYNAP [1] and the interactive GRAPE [2] as the three-dimensional post-processor for NIKE3D [3], DYNA3D [1], TACO3D [4], and GEMINI [5]. If desired, TAURUS can now also be used as a post-processor for the two-dimensional programs NIKE2D [6,7], DYNA2D [8], TACO2D [9], and TOPAZ [10]. TAURUS combines the GRAPE code for plotting geometry and contours with the capability contained in the DYNAP program. A simplified command structure has been adopted to replace the GRAPE command structure. This new structure is similar to, but not identical to that used in ORION [11] and MAZE [12]. GRAPE and, therefore, TAURUS to some extent, has its origins with the work of Christiansen* and his code MOVIE.BYU [13] which was brought to LLNL in 1973 and extended into SAMPP [14].

*Professor of Civil Engineering, Brigham Young University, Provo, Utah.

COMMANDS

Commands in TAURUS are very simple and understandable, and only a small subset of the available commands are needed to effectively plot data. Like ORION, TAURUS has three phases:

- phase 0: initialization,
- phase 1: geometry display with contouring,
- phase 2: time history processing.

with the last phase nearly identical in both codes. Available commands are listed in Fig. 1. Knowledge of the phase 1 commands marked with an asterisk are sufficient to do geometry and contour plotting. This constitutes considerably less knowledge than was needed to use GRAPE.

Commands common to Phases 1 & 2	Phase 1 commands			Phase 2 commands
TV	DAM*	RESO	MIXED	PHS1
T or END	M*	TRANS	POOR	ELEMENTS
HEAD	P	XTRANS*	RAXY	NODES
PLOTS	DMS	YTRANS	ARXY	MATLS
C	DPS	ZTRANS	RAYZ	GATHER
TTY	CMN	CENTER	ARYZ	EXPDATA
CFILE	TIME*	WHERE	RAZX	ETIME
RJET	STATE	FOLLOW	ARZX	NTIME
	DSF	SUMM	R	MTIME
	DSFS	ANGLE	RMC	NRTIME
	XSCALE	ZMIN	MOVIE	GTIME
	YSCALE	ZMAX	ANIMATE	PRINT
	ZSCALE	SLICE	PAUSE	ASET
	UDG	DIST*	TRPT	OSET
	CONTOUR*	TRIAD	CSIZE	ASCL
	WV	VIEW*	PHS2	OSCL
	WC	DRAW	COLOR	SMOOTH
	RANGE	PVIEW	CB	
	NUMCON	PDRAW	CM	
	RX	NDPLT	CP	
	RY	EXPLODE	CFM	
	RZ	ESF	CFP	
		PIVOT	FLAT	
		STICK	SMOOTH	
		BEAM	UNIFORM	
		RESTORE	DIFM	
		REVERSE	DIFP	

FIG. 1. Commands in TAURUS.

EXECUTION

TAURUS is a public file on the CRAY and 7600 computers and may be executed by typing

TAURUS C=**cfℓ** G=**ptf** S=**sfℓ**

where

cfℓ = command file for batch execution,
ptf = binary plot file,
sfℓ = save file for teletypewriter commands.

Because the file, **cfℓ** is not required, and the default file name of **sfℓ** is TSAVE, the execution line

TAURUS G=**ptf**

is fully acceptable. The optional command file, **cfℓ**, is processed until either a termination command, TTY command, or an end-of-file is found. In the latter case, control is returned to the teletypewriters. The first line in the command file must be the user's box number and identification, i.e.,

BOX ann identification

for the graphics file, and the first entry on the second line must be a TMDS (Television Monitor Display System) number. If the first line of the input is blank, no graphics file is created. If no TMDS is needed, then a zero should be used.

On VAX computers, type

RUN TAURUS

TAURUS will then prompt for the balance of the execution line.

COORDINATE SYSTEMS

TAURUS assumes that the user is sitting in a fixed position looking at the model. The model is then rotated, zoomed in on, and translated upon the user's command. To fully utilize TAURUS, the coordinate systems must be known and understood. For the user looking at a TMDS display, the positive X axis is always horizontal from left to right, and the positive Y axis is vertical from bottom to top. The Z axis is normal to the screen and for rotations and translations the Z axis is positive coming out of the screen (a right handed coordinate system). The origin is at the center of the screen, see Fig. 2. The triad plotted in the lower left hand corner of the display is optional via the TRIAD command. This triad tracks the model's coordinate system. To view the model, it is placed in the frustum of vision (in our case a truncated pyramid) also known as the truncated view volume. The tip of the pyramid is at the origin and is the location of the viewer. The amount of perspective in the final image is controlled by the included angle at the tip of the pyramid. Portions of the model which extend beyond the frustum are clipped or removed prior to display. If we view the frustum from above, looking down the Y axis, we would see the view shown in Fig. 3, for our cube in Fig. 2. The Z-minimum, ZMIN, and the Z-maximum, ZMAX, planes are also under user control if the defaults are not satisfactory.

During initialization, the code finds the range of the model in X, Y, and Z, then translates the model's origin to the center of ranges. The frustum of vision's included angle is set to 45 degrees. The distance from the viewer to the model's origin is then set so that the model will be totally inside of the frustrum and no clipping will occur as opposed to the model shown in Fig. 4 which is clipped. The view in Fig. 5 shows the cube of Fig. 4 as it would be displayed. The model may be displayed with either the DRAW (display all lines) or VIEW (remove the hidden lines or make a continuous tone image) commands. When run interactively, TAURUS interrogates the user for a box number and a TMDS number and proceeds to initialize. TAURUS defaults the many quantities that GRAPE requested as input while initializing. However, these defaults may be changed later if desired.

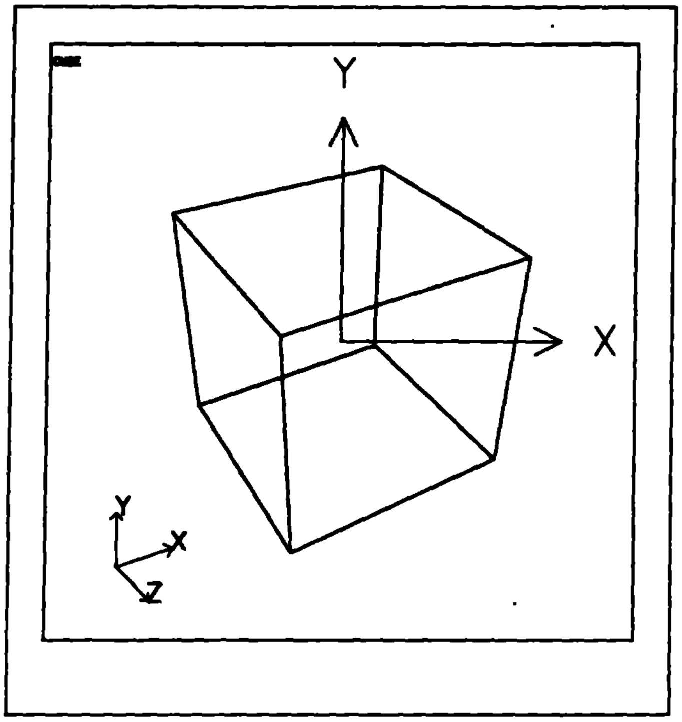


FIG. 2.

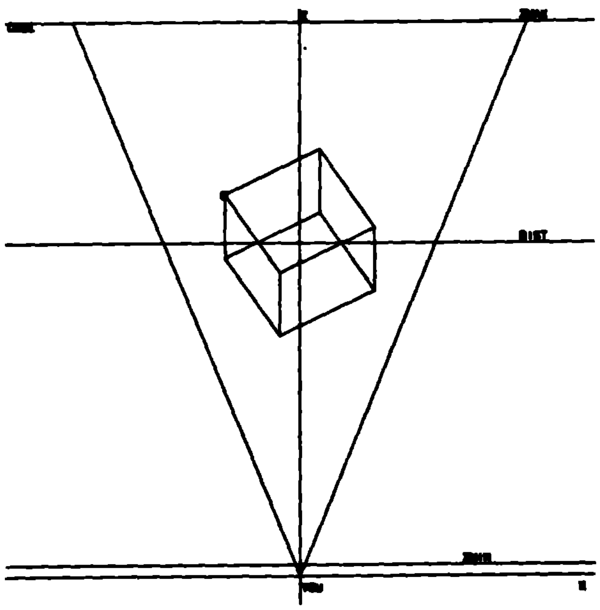


FIG. 3.

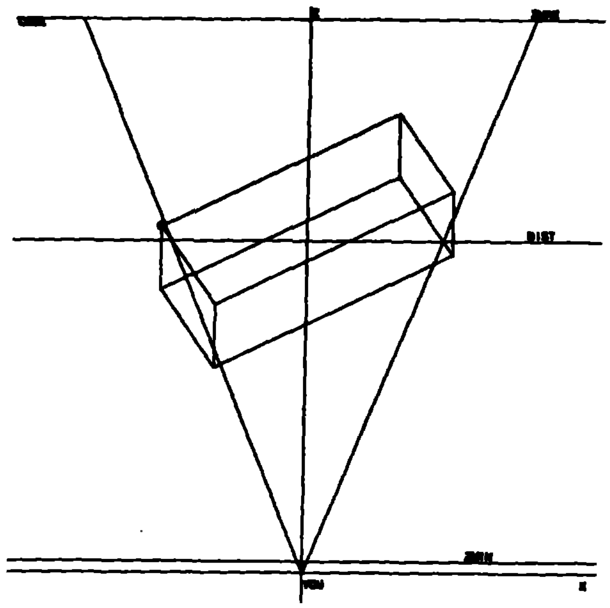


FIG. 4.

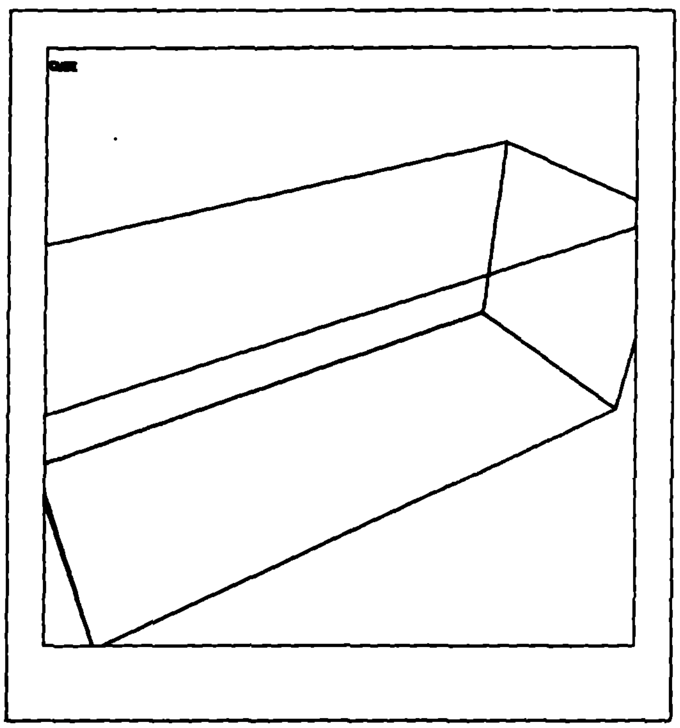


FIG. 5.

Translation of a model is accomplished two ways. In the model's coordinate system the TRANS command is available for specifying a new origin, and in the screen's coordinate system the XTRANS, YTRANS, and ZTRANS commands are available for translating the model relative to the origin. Figure 6 shows our cube again, one unit on an edge, whose origin is at one of the corners. The "0" in Fig. 7 corresponds to the model's origin. If we perform no translation, the origin of the model's system in X and Y will correspond with the screen's (display) origin (the center of the screen, Fig. 2). The Z values for the origin of the model will be the distance value which can be reset by the DIST command. The distance value in Figs. 6 and 7 is three for our unit cube. When the origin is translated to the center of the cube we have the views in Fig. 8 and 9. This is the default value, the center of each coordinate range, for positioning the model the first time the display overlay is entered. In Figs. 10 and 11 we show the cube with the origin translated to the point (0.7,0.7,0.7). Note that the model's coordinate system is assumed to be right-handed, while the frustum of vision's coordinate system is left-handed (the positive sense of the Z axis is reversed).

Rotations of the model are always performed with respect to the screen coordinate system. The point about which the model is rotated is the current origin. If the cube in Fig. 10 is rotated 30 degrees about the Y axis, the view in Fig. 12 is generated. Comparing the view of the cube in Fig. 11 with that in Fig. 13 we see that the model has been rotated about the Y axis.

To accomplish the actual rotations, the coordinates of each point are multiplied by a 3x3 direction cosine (DC) matrix. Mathematically the new values of x, y, and z (the prime values) are given by Eq. (1).

$$[x' \ y' \ z'] = [DC] \ [x \ y \ z] \tag{1}$$

Initially DC is an identity matrix. The command RESTORE will reset DC to the identity matrix when executed, since the rotate commands RX, RY, and RZ will change the values of DC according to the angle given. For rotation about the X axis of angle α (+ α is counterclockwise, see Fig. 14A), the elements of the DC matrix are:

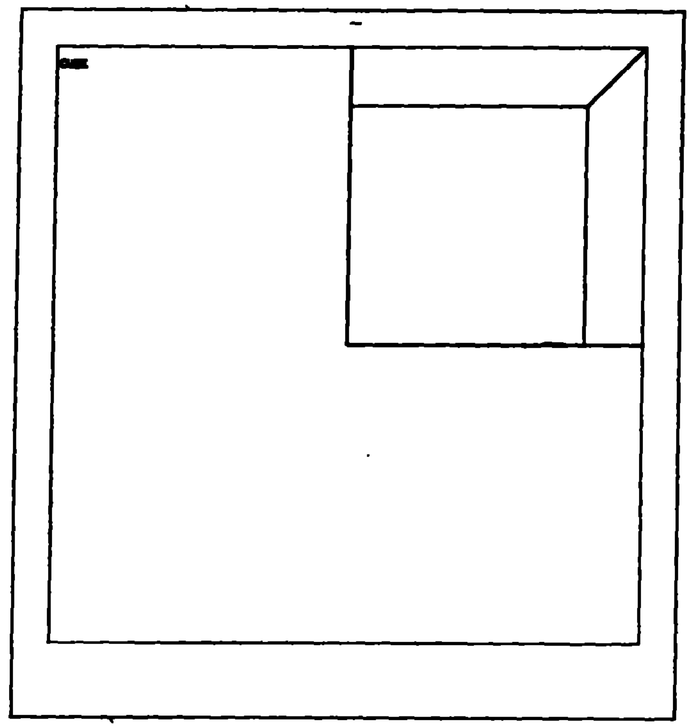


FIG. 6.

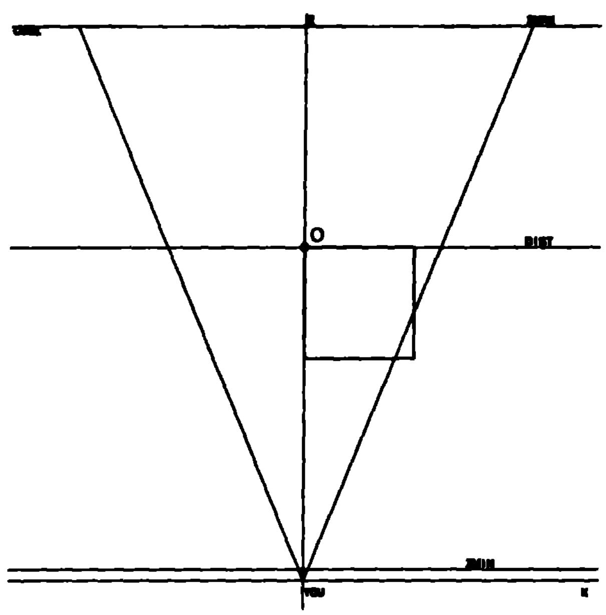


FIG. 7.

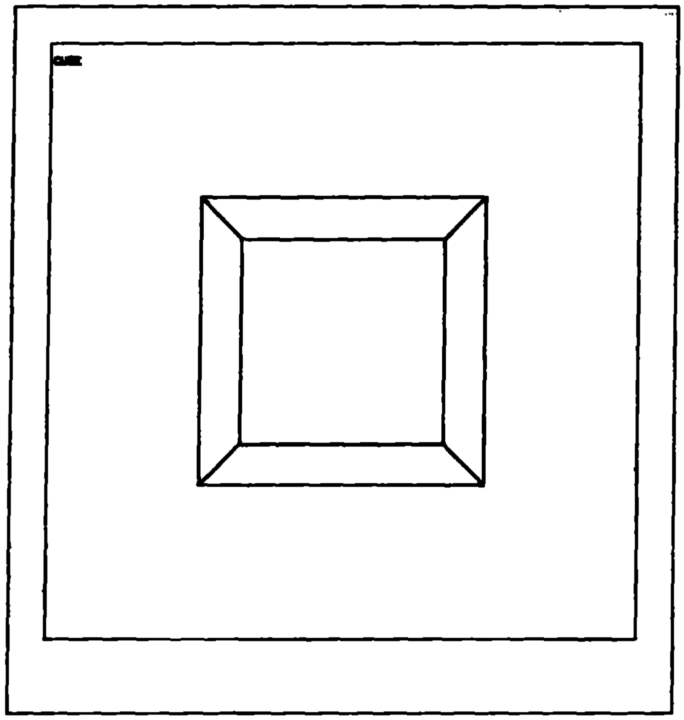


FIG. 8.

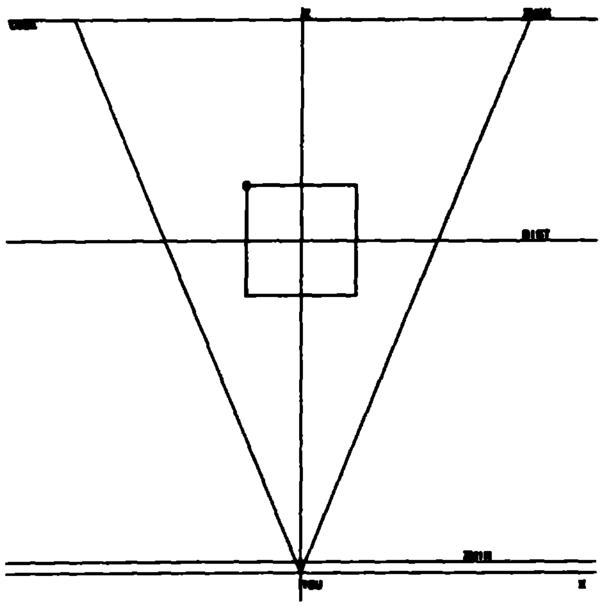


FIG. 9.

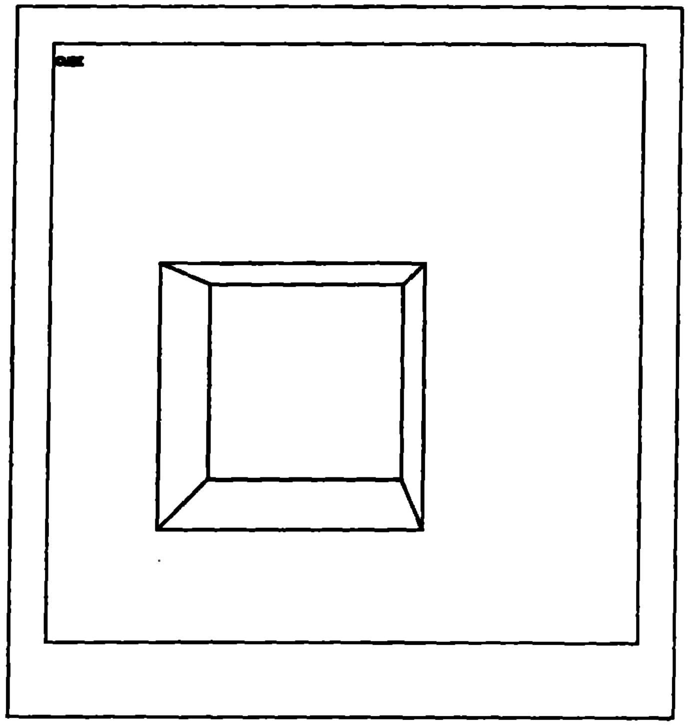


FIG. 10.

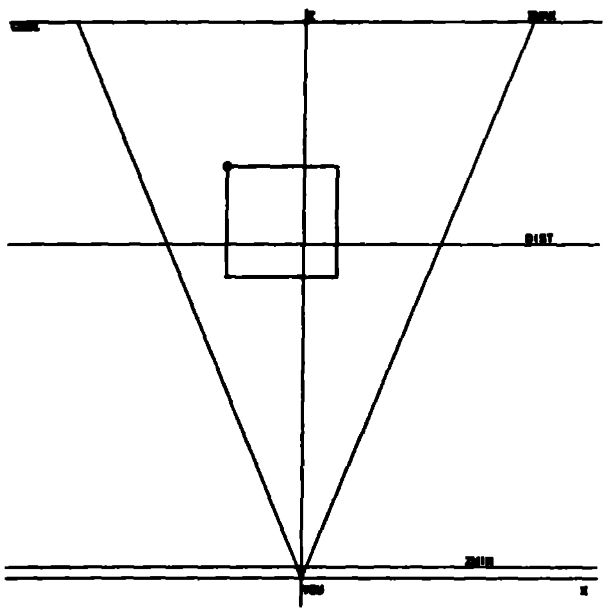


FIG. 11.

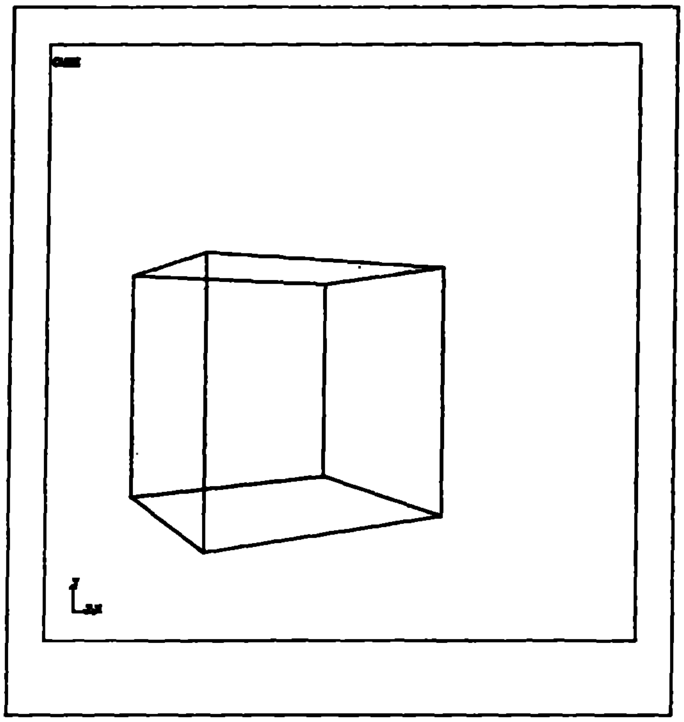


FIG. 12.

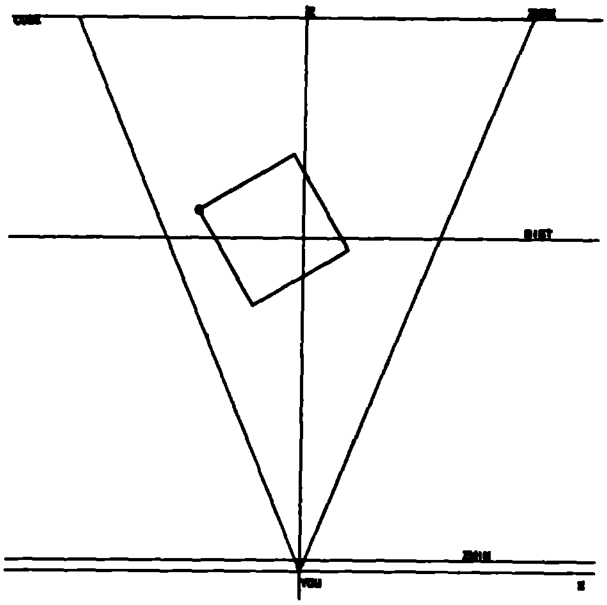


FIG. 13.

$$[DC_x] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\alpha & s\alpha \\ 0 & -s\alpha & c\alpha \end{bmatrix} \qquad (2)$$

where $s\alpha$ is the $\sin(\alpha)$ and $c\alpha$ is the $\cos(\alpha)$. For a rotation about the Y axis of β (for positive β see Fig. 14B) the DC matrix is:

$$[DC_y] = \begin{bmatrix} c\beta & 0 & -s\beta \\ 0 & 1 & 0 \\ s\beta & 0 & c\beta \end{bmatrix} \qquad (3)$$

where $s\beta$ is the $\sin(\beta)$ and $c\beta$ is the $\cos(\beta)$. Similarly, for a rotation of γ about the Z axis (see Fig. 14C) the DC matrix is:

$$[DC_z] = \begin{bmatrix} c\gamma & s\gamma & 0 \\ -s\gamma & c\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (4)$$

where $s\gamma$ is the $\sin(\gamma)$ and $c\gamma$ is the $\cos(\gamma)$.

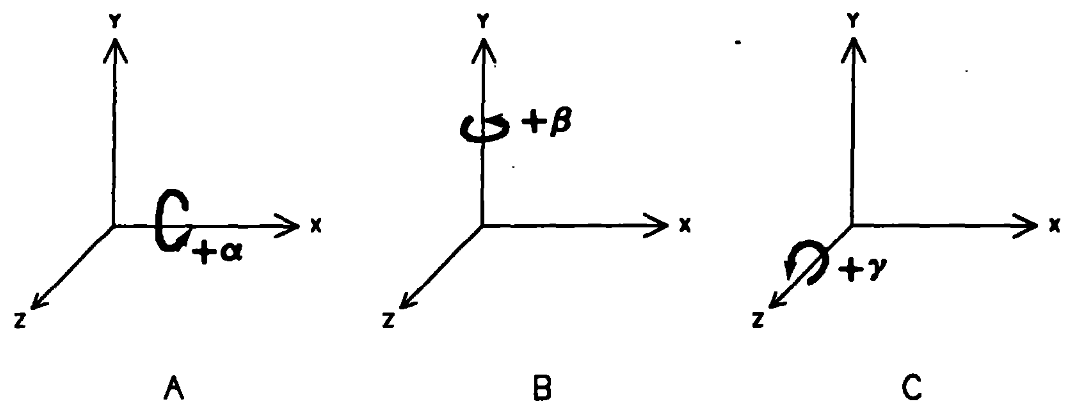


FIG. 14.

When more than one rotation of the model is applied, these DC matrices are concatenated prior to multiplying the coordinates. The latest DC matrix always post-multiplies the previous result. For example if the model were rotated about the X axis by α, and about the Y axis by β, the resulting DC matrix would look like:

$$[DC_{xy}] = \begin{bmatrix} c\beta & 0 & -s\beta \\ s\alpha s\beta & c\alpha & c\alpha c\beta \\ c\alpha s\beta & -s\alpha & c\alpha c\beta \end{bmatrix} \tag{5}$$

If the model is then rotated about the Z axis by γ the result is:

$$[DC_{xyz}] = \begin{bmatrix} c\beta c\gamma & c\beta s\gamma & -s\beta \\ s\alpha s\beta c\gamma - c\alpha s\gamma & c\alpha s\beta s\gamma + c\alpha c\gamma & s\alpha c\beta \\ c\alpha s\beta c\gamma + s\alpha s\gamma & c\alpha s\beta s\gamma - s\alpha c\gamma & c\alpha c\beta \end{bmatrix}$$

Any number of rotations about any of the three axes in any order can be described by three or less rotations. The first rotation is about the X axis followed by a rotation about the Y axis and if needed a rotation about the Z axis. In other words we can solve the $[DC_{xyz}]$ for α, β, and γ. This is accomplished in the TAURUS by the WHERE command. When WHERE is executed β is determined by taking the arcsine of minus the value in $\{1,3\}$ (row 1, column 3) of the current DC matrix. Assuming the value of β is not ± 90 degrees then α and γ may be determined from either of the elements of the first row and third column of the DC matrix. When β is ± 90 degrees then α and γ are not independent. We assume that γ is zero and solve for α using the $\{3,1\}$ term of DC.

Another option, called ANGLE, may also be described using our unit cube. The amount of perspective in the displayed image is controlled by the included angle of our frustum. The default for this angle is 45 degrees. The useful range is from 0.1 degrees to 179 degrees. Values of zero and equal to or greater than 180 degrees cause mathematical problems and should not be used. Usually the default is fine, but we demonstrate this option for those who need it. If we take the view of our cube in Figs. 2 and 3 and change the angle from 45 to 90 degrees we obtain the views in Figs. 15 and 16. Notice that in Fig. 16 the cube is the same size as it is in Fig. 3 but the view in Fig. 15 is much smaller than Fig. 2. We have essentially changed the lens on our viewing camera from "normal" to "wide angle." The perspective is severe and can be easily seen when the vertical edges of the cube are compared. If the view were displayed with no perspective transformation performed upon the values, i.e., an orthogonal projection, these lines would remain parallel. Alternatively we can reduce our perspective by narrowing the angle. In Figs. 17 and 18 we have used a 30 degree field. Using the camera again we have now put on a "telephoto" lens. The cube appears larger and the vertical lines appear more parallel.

As the DISTANCE value is increased it becomes closer to the ZMAX plane's value. The range or spread between the ZMIN and ZMAX planes is used by the hidden surface processor to scale the Z values. Since visibility in the X, Y plane (the screen), is a function of Z we try to keep a maximum of significant bits around for the Z values.

Besides being used by the code to scale the Z data, the ZMAX and ZMIN planes may be used to clip the data. Taking our standard view of the cube, Fig. 2, we can bring the ZMAX plane in from five to 3.5. The resulting images are shown in Figs. 19 and 20. This use of the ZMAX plane to clip the data may be advantageous to the hidden surface processor if we use the plane to remove data we know we cannot see. Similarly the ZMIN plane may be moved. In Figs. 21 and 22, we have set the ZMIN plane to 2.5 and displayed the image with the DRAW command. If the VIEW command instead were used, then the Fig. 23 is generated. Our cube appears to have had a corner clipped off and appears hollow.

If the model is defined as a collection of hexahedrons (bricks) the SLICE command may be used. SLICE also allows the user to set the position of the ZMIN and ZMAX planes and, when used in conjunction with the VIEW command, clips all of the bricks and replaces the clipped portions with a new polygon. This replacement is sometimes called "capping". When continuums are being modeled in three dimensions this capping makes the model appear to be solid. We have used SLICE to produce the image shown in Fig. 24. This is a very useful option of the code and will be discussed further when the display of results of finite element calculations is covered.

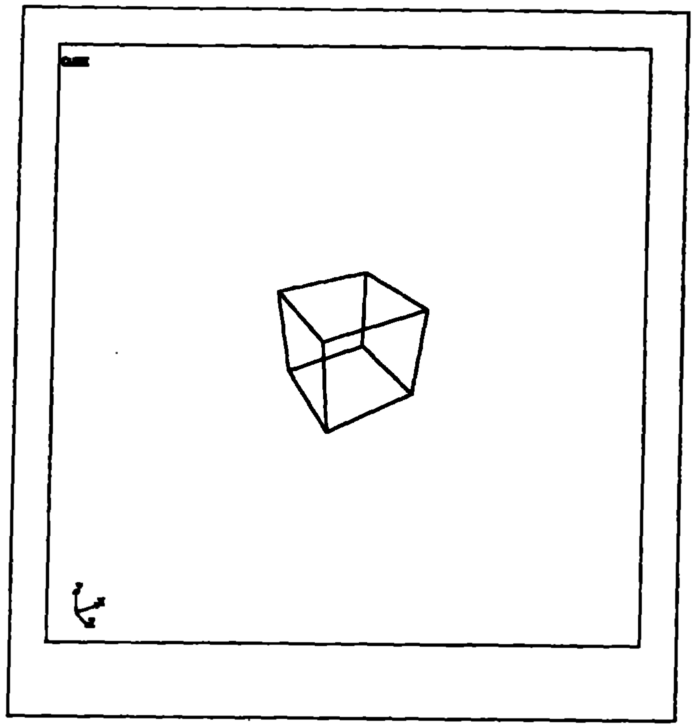


FIG. 15.

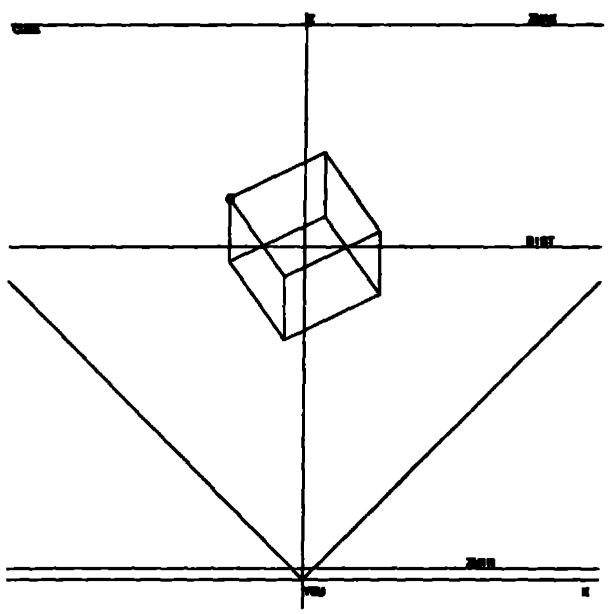


FIG. 16.

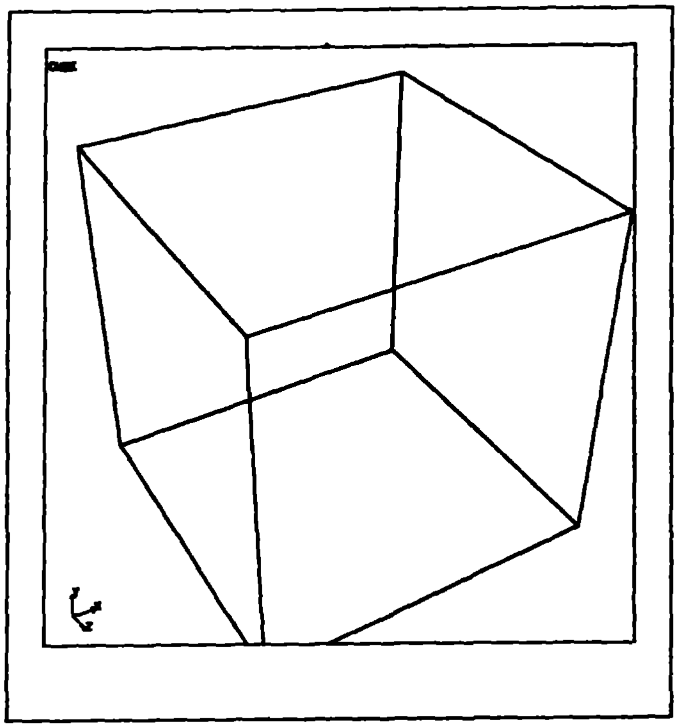


FIG. 17.

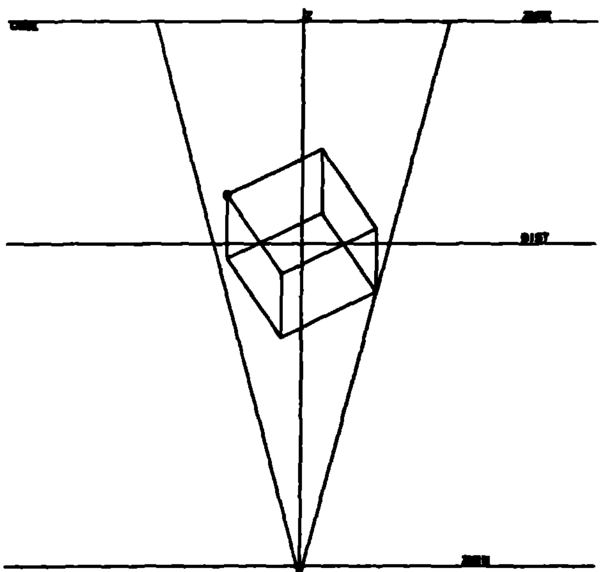


FIG. 18.

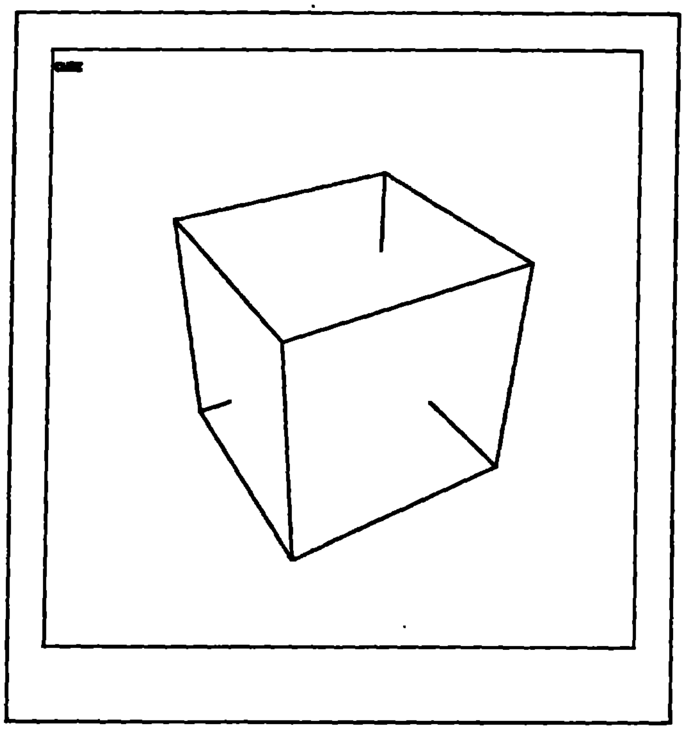


FIG. 19.

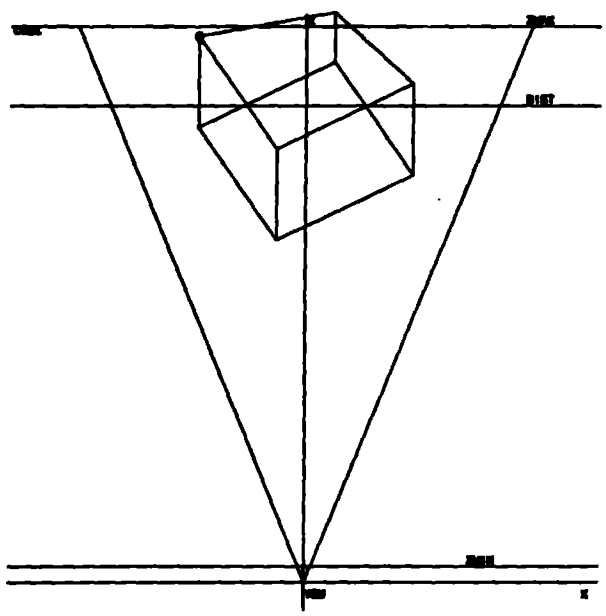


FIG. 20.

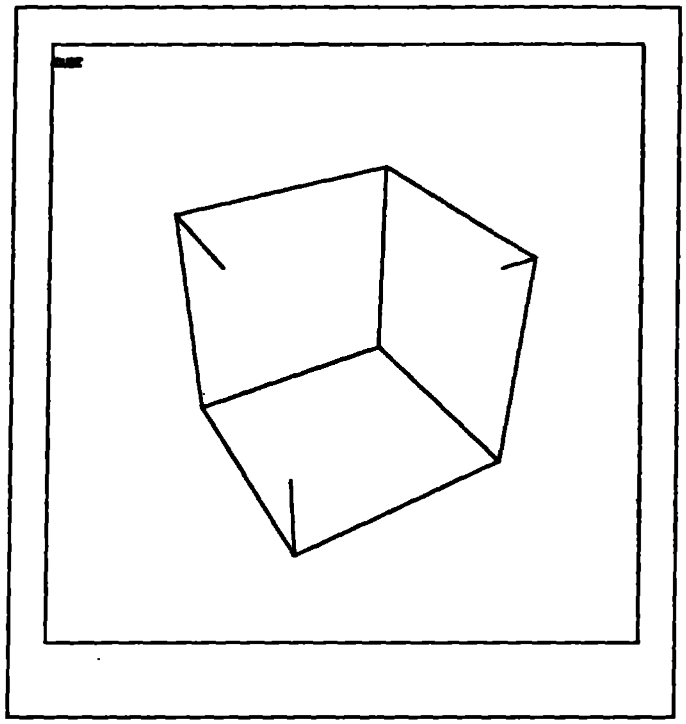


FIG. 21.

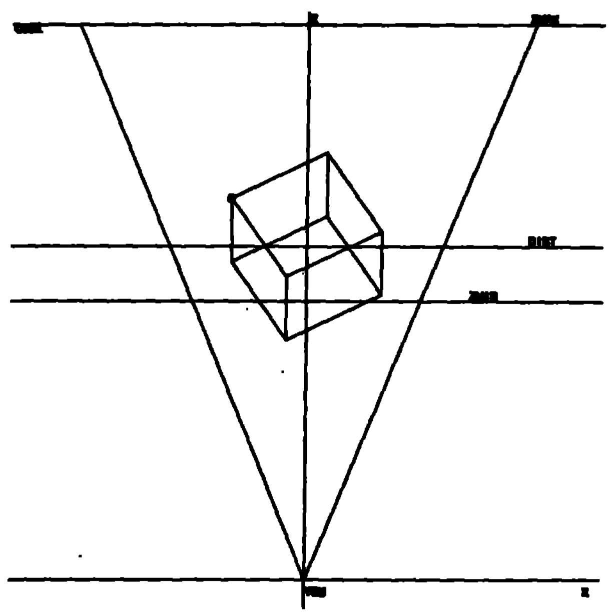


FIG. 22.

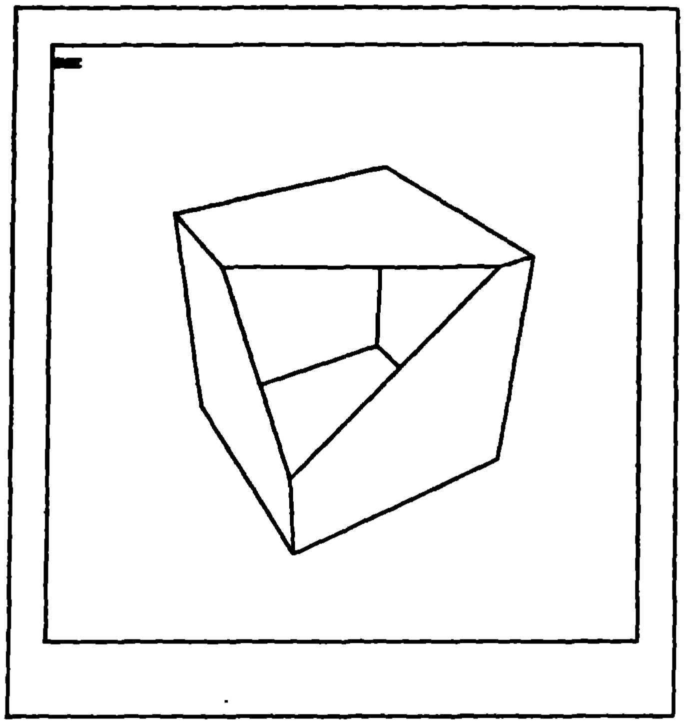


FIG. 23.

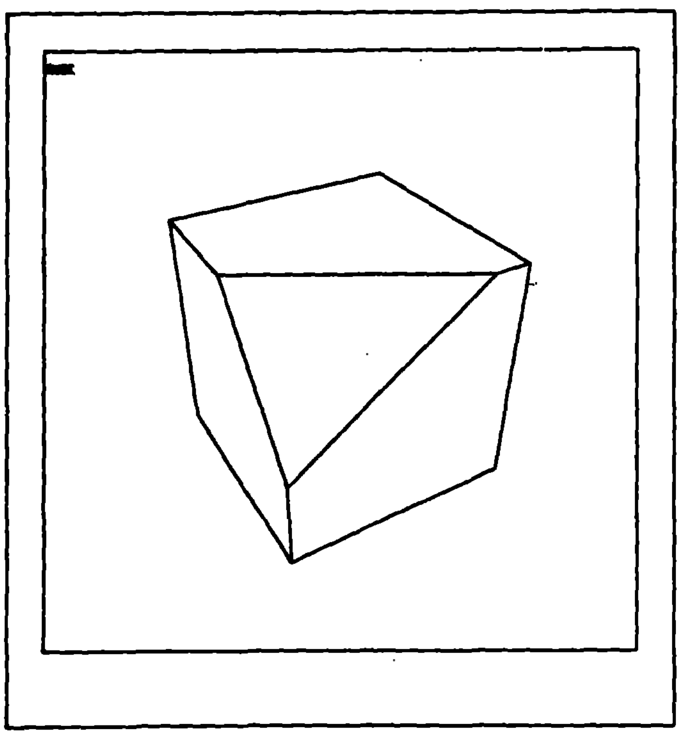


FIG. 24.

Another option of TAURUS is the NDPLT command. This command places node numbers on the plots and is useful for debugging geometries. The size of the characters is under user control. The default size allows 75 characters to span the image horizontally. The command CSIZE allows the user to specify the number of characters per line. When the model has many nodes, the size of the characters may be reduced so when the view is magnified, the numbers do not interfere with each other. Our cube is shown in Fig. 25 using the other extreme. We have set CSIZE to be 30 so the numbers will be large.

There are occasions when the model being displayed is very long and slender or otherwise ill-suited for display on a square output format. For this case and others TAURUS can scale the initial model coordinates in either the X, Y, or Z directions. The commands XSCALE, YSCALE, and ZSCALE allow the user to specify the scale factors to be applied. We have used this option to produce the views shown in Figs. 4 and 5. An XSCALE value of 2.5 was given.

This ends the discussion of the major commands which have something to do with the coordinate systems. The next portion will describe commands which have to do with materials and parts.

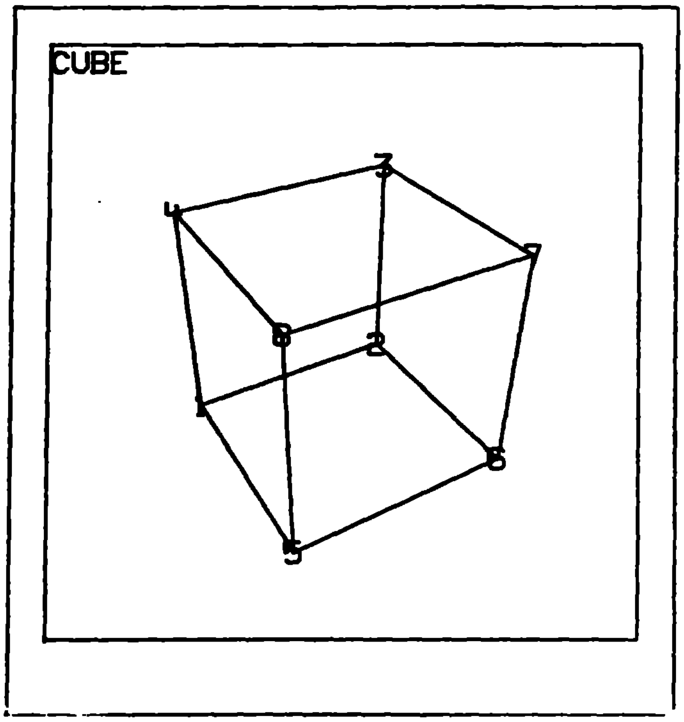


FIG. 25.

MATERIALS AND PARTS

There are several commands in TAURUS that require material or part numbers. Parts or materials are defined, for our purposes, to be collections of individual elements which represent a unit. The distinction between parts and materials occurs whenever the reflect options RAXY, RAYZ, and RAZX are used. These latter options preserves the number of materials but double the number of parts. Hence, reflected parts have part numbers NP+1 to 2NP where NP is the number of parts prior to the reflect command. Part i ($1 \leq i \leq NP$) reflected is assigned part number i+NP, but material i now includes both parts i and NP+i. The capability to display by material number is a convenience that eliminates the necessity of computing part numbers.

We have subdivided our cube into eight cubes and displayed it without the hidden lines removed in Fig. 26. Furthermore, we have also specified that one brick has a different part number. A view of part number one is shown in Fig. 27. The command PVIEW was used to generate this view. We have not bothered to show the second part but the PVIEW command will display one part per image for all parts in the model. PDRAW is its non-hidden line removal counterpart.

EXPLODE is a command which allows relative motion of the various parts. The user specifies a translation vector for a subset of parts and a general scale factor to be applied to the vector's components. Each part is then translated by this scale factors times the vector after being rotated and translated by the rotation and translation commands, respectively. To demonstrate this we have translated part 2 of our model with a vector of (0.5,0.5,0.5) using a scale factor of 1.0, see Fig. 28. The vector given was relative to the model's coordinate system. The RESTORE command will zero all vectors specified by this command.

EXPLODE has been very useful when displaying models of several parts. The exploded view can be generated to show the viewer the relative positioning of the various components. This command can be used with the ANIMATE command to provide an animated sequence.

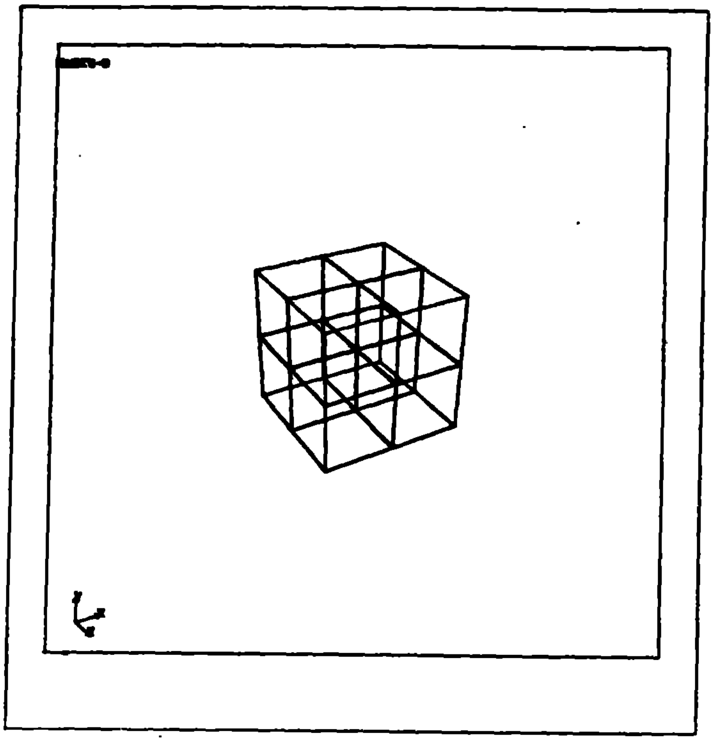


FIG. 26.

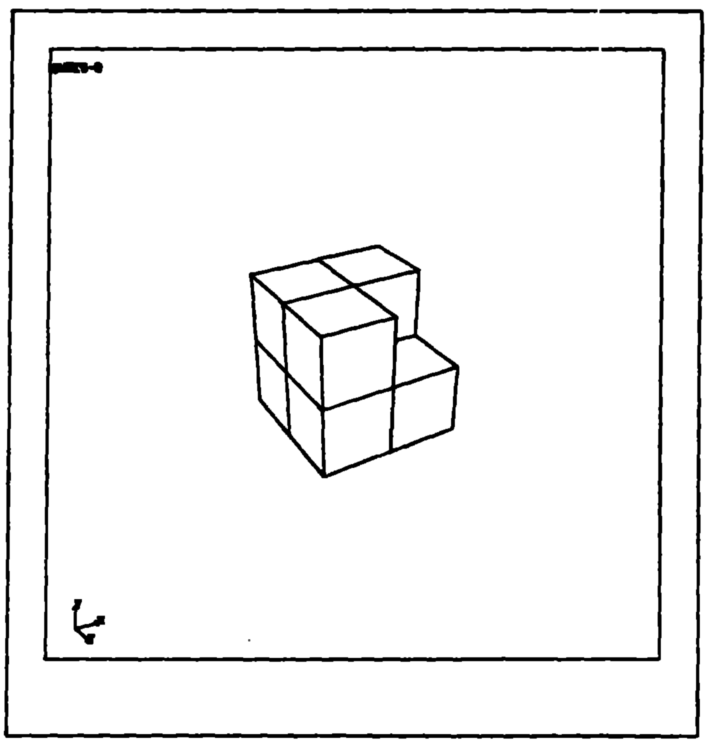


FIG. 27.

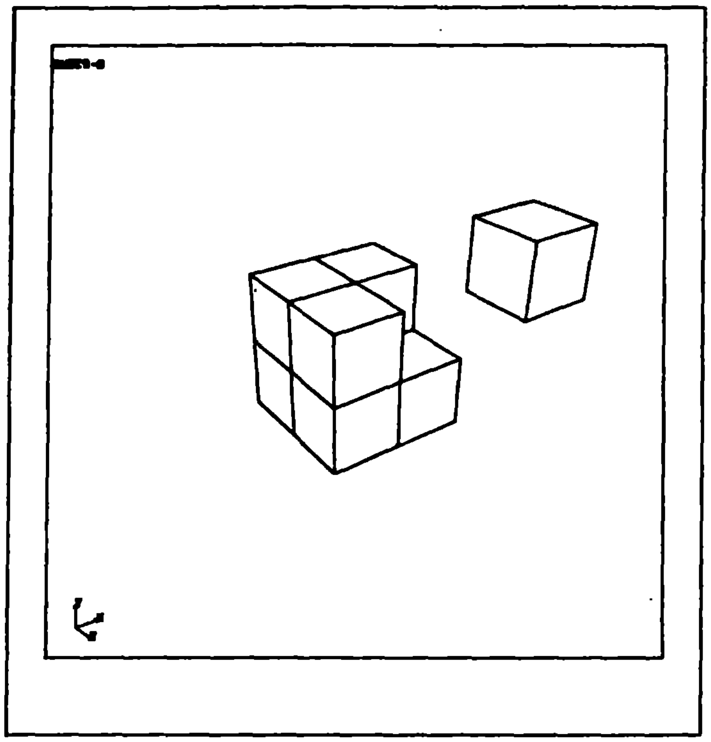


FIG. 28.

Another command which is dependent on having more than one part is PIVOT. PIVOT allows the user to specify rotation of each part about a relative origin. We have demonstrated this in Fig. 28. Part 2 has been rotated 45 degrees about the Y axis with the relative origin set at (1.0,1.0,1.0). Since the part is also undergoing a relative motion (from EXPLODE), the origin for the rotation was also translated. To see this effect, Figs. 28 and 29 have been superimposed in Fig. 30. We see that the rotation has been about the upper right hand corner of the cube.

To summarize the transformation applied to the model are:

1. Rotation (from the RX, RY, and RZ commands)
2. Translation (from the translate commands TRANS, XTRANS, YTRANS, and ZTRANS).
3. For each part
   3.1 Relative translation (from the EXPLODE command).
   3.2 Rotation about a relative origin (from the PIVOT command).

The next section of the manual describes the display of data calculated by the analysis routines.

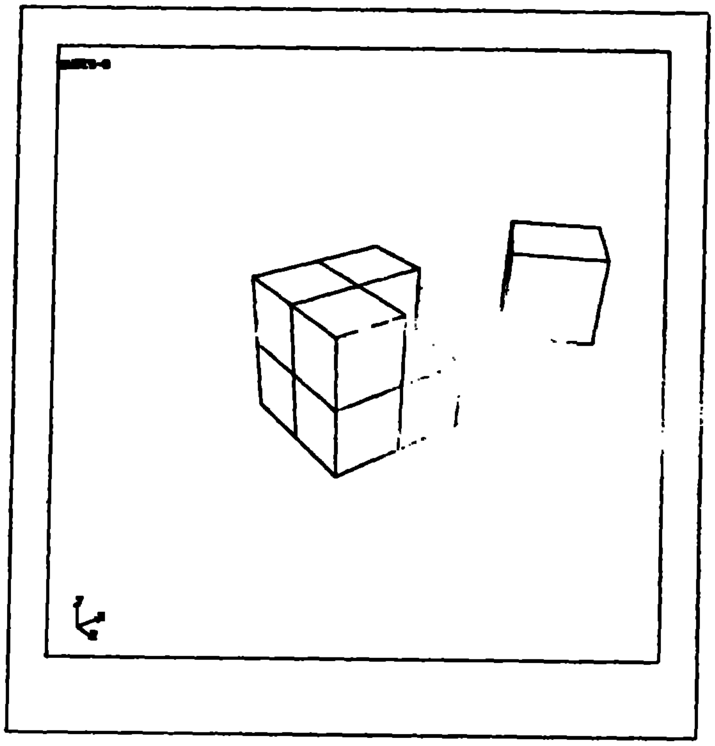


FIG. 29.

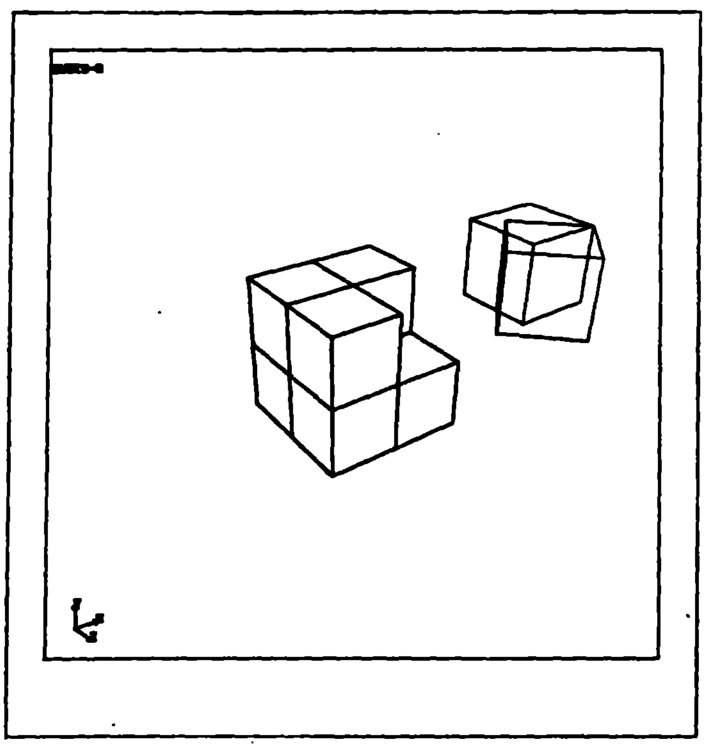


FIG. 30.

DISPLAY AND RESULTS

Currently two types of data from the analysis routines may be displayed. The first is scalar data (one variable quantity at each node), and the second is vector data (displacement). These scalar quantities may be values of pressure, temperature, components of the stress/strain tensor, etc. Scalar data is obtained from the elemental values (such as centroidal or Gauss point stresses) by averaging at the nodes. Use of the CONTOUR command causes contour lines representing the value of the scalar function to be drawn on the surface of the model. Our eight cubes have scalar value assigned to each of the 27 nodes with a range from 1 to 10.

The contour lines shown in Fig. 31 are on 1.0 increments and labeled with alphabetical characters. The user may default the contour levels and the number of contours (default is 9) or he may set these quantities with the RANGE and NUMC commands, respectively.

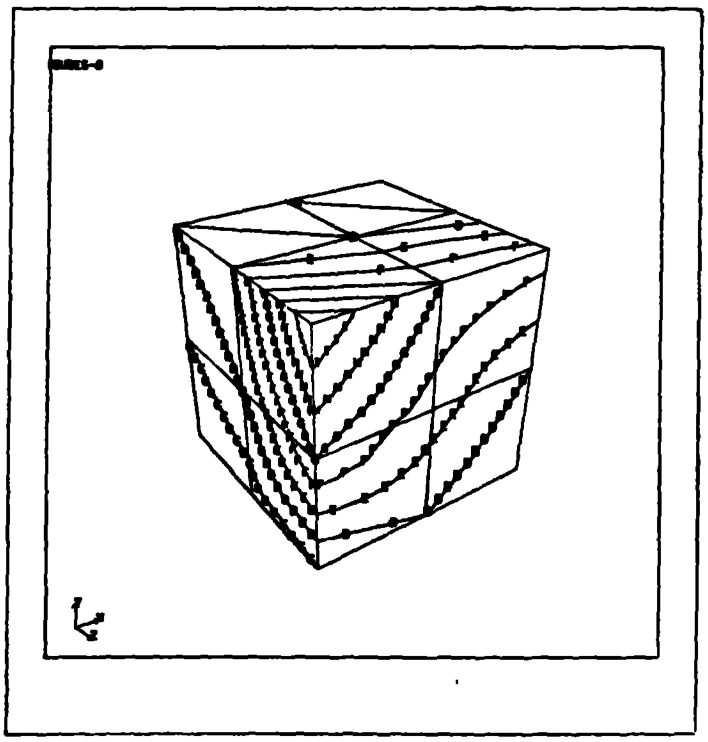


FIG. 31.

Since the results of the analysis of solid objects extends throughout the model, a method for viewing these internal values needs to exist. The SLICE command as previously mentioned is one alternative. In Fig. 32 we display the cube with the corner sliced off. Interpolated to the capping polygon are the scalar values which are then plotted with the contour lines. Figure 33 shows, with the output of the WHERE command, the position of our model versus the clipping planes.

A second example is shown in Figs. 34 and 35. We have a flat plate in Fig. 34 and the contours of a scalar function plotted on it in Fig. 35. The range this time is 0.0 to 1.0 by increments of 0.1. The warp and view command, WV, allows the user to convert the scalar function to a vector by specifying vector components for the scalar. This option works well for models which are two-dimensional, or for models which have large planar areas. We have warped the flat plate with a vector of (0.0,0.0,10.0) and show the image in Fig. 36. We can also display the modeled warped and contoured as shown in Fig. 37 with the WC command.

Sometimes it is useful to plot the original shape of the model on the top of the deformed mesh. Two options are available for this over-plotting. Assume our eight cubes have a displacement field that shrinks the model. If we choose to display the original shape with points at the original nodal positions and draw the deformed shape, we obtain the view found in Fig. 38. The alternative is to draw the original geometry with solid lines as shown in Fig. 39. Of course, for many models, the original geometry is not important or is self evident so the option of not plotting it is the default.

ANIMATION

The ANIMATE command allows specification of the type and amount of animation. It is a command that has many options. Several of these options are functions of previous commands and should not be activated unless their related commands have been used. ANIMATE is very useful for a few frames viewed on the TMDS or for movies, and users should become familiar with this capability. The first argument is the number of frames, n, followed by zero, one, or more options with the terminate command ENDA at the end.

If a vector function (displacement) has been read, the ANIMATE option VIBR allows the user to specify a value for the number of cycles of vibrations per n frames. If the displacement state is a normal mode of vibration, ANIMATE then changes the scaling factor (SCF) sinusoidally, producing a sequence in which the model vibrates.

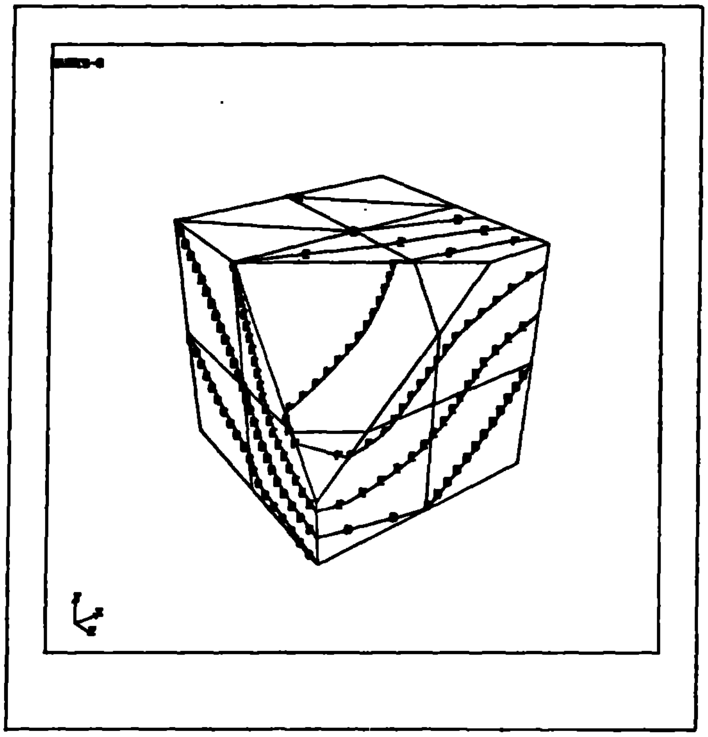


FIG. 32.

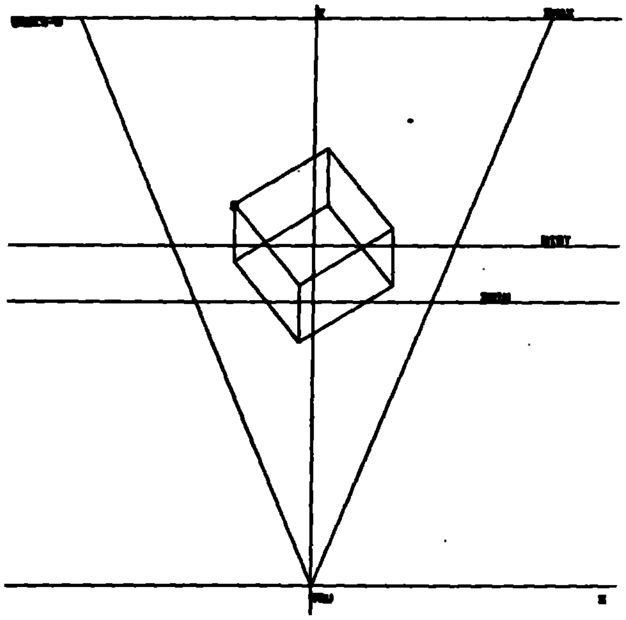


FIG. 33.

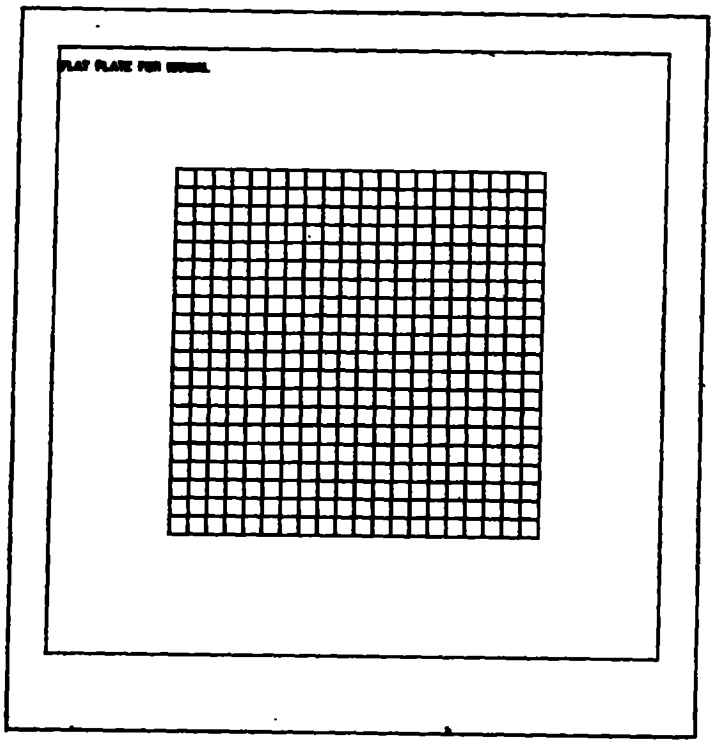


FIG. 34.

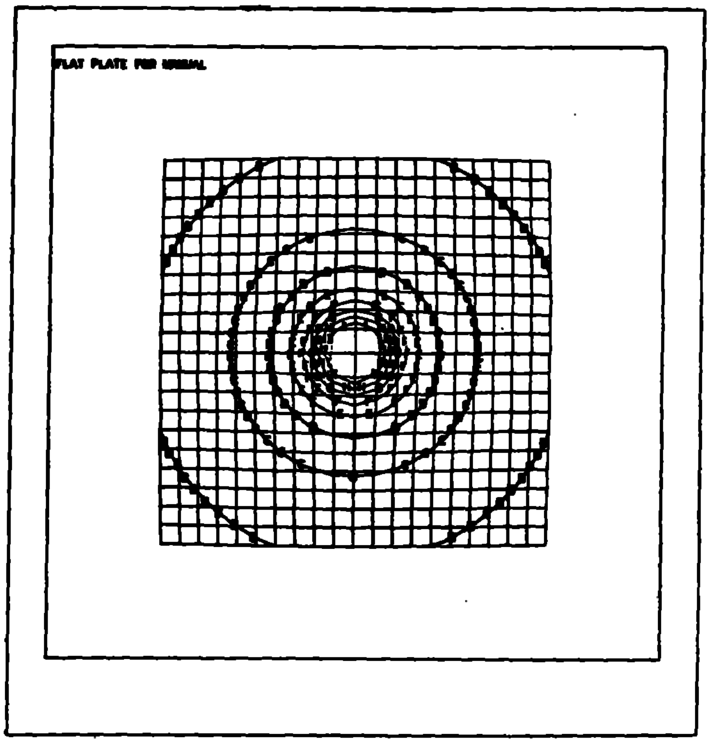


FIG. 35.

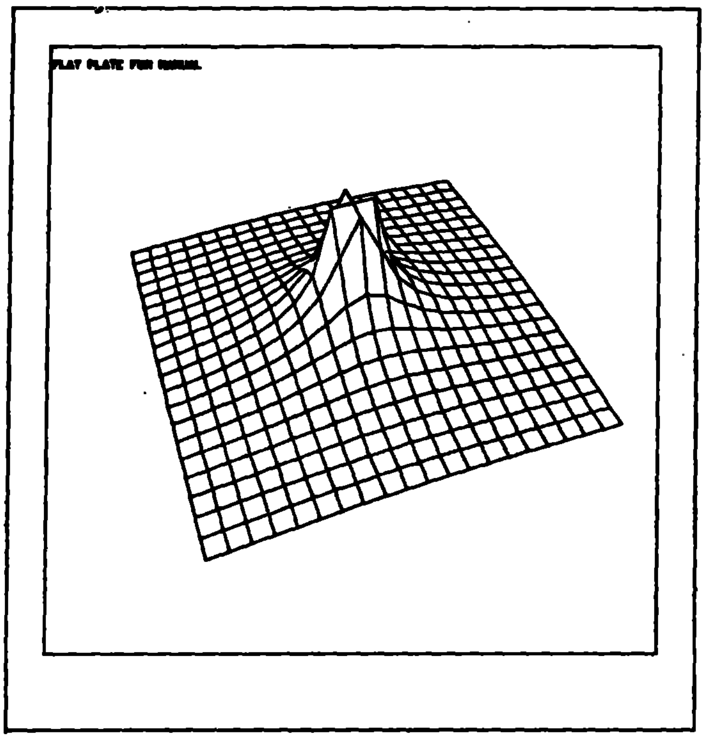


FIG. 36.

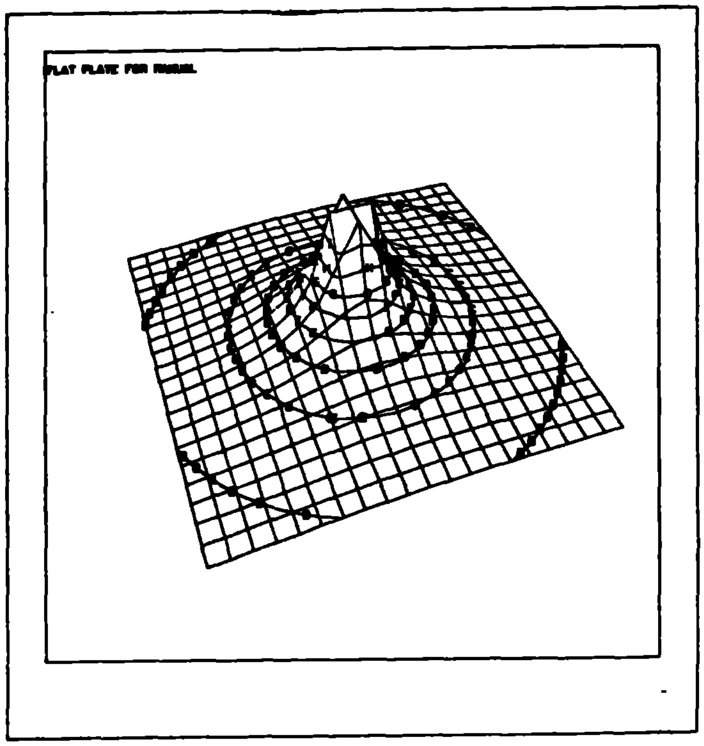


FIG. 37.

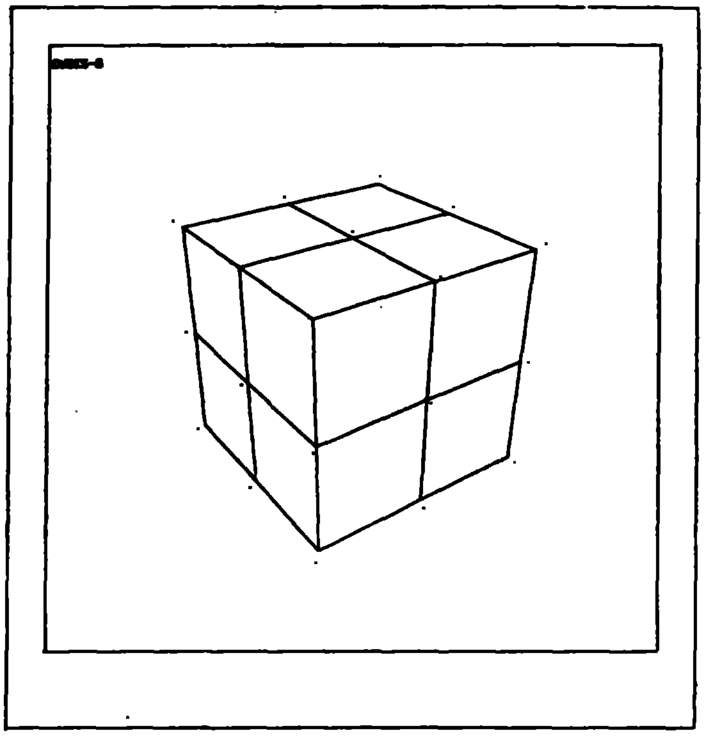


FIG. 38.

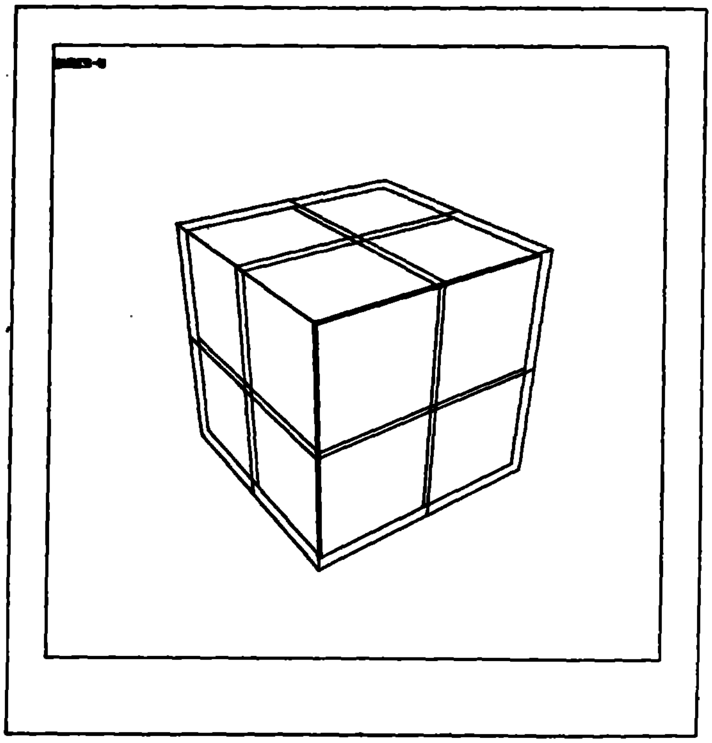


FIG. 39.

The ANIMATE option TRNS allows animation changes to the translation of the model. This will take place over the number of frames specified. We have used our single cube model to demonstrate this feature. The cube is placed in the upper right hand corner of the screen using the command TRANS. In ANIMATE, if we specify that the total translation change in five frames is to be 1.25,1.25,1.25, that is, translate the origin of the model from where it is at (x,y,z) to (x+1.25,y+1.25,y+1.25), Fig. 40 is obtained. Here, the original position with the five frames plotted on the same frame is shown. The motion was not uniform from the original position to the final. All animation is performed using what has been termed an "elevator function". The model begins accelerating to the halfway point then decelerates to its final position. This type of smooth animation has been found to be much more pleasing to the viewer of movie sequences. The alternative is to move at a uniform rate. The infinite acceleration at the beginning and infinite deceleration at the end of the motion is uncomfortable to watch.

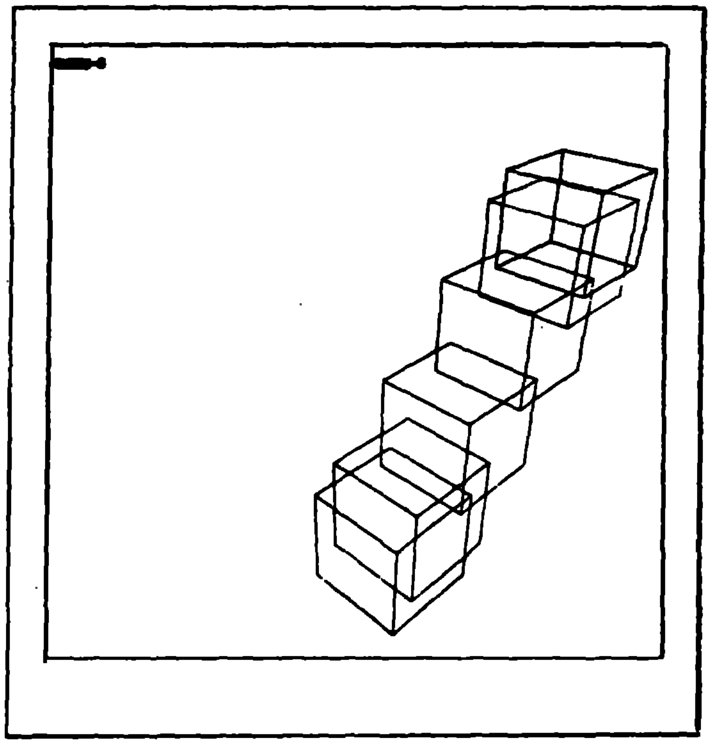


FIG. 40.

The ANIMATE option ROTA permits rotation of the model. The user specifies the rotation about the X, Y, and Z axes. Care should be taken here since finite rotations do not add as vectors; therefore, to avoid unknown results the best scheme is to rotate about one axis at a time. We demonstrate rotating about the Z axis by 90 degrees in Fig. 41. Again, the original position plus the five frames generated by the ANIMATE command have been superimposed.

Changing the value of the distance to the origin may be accomplished by the DELD option of ANIMATE. In Fig. 42 we have reduced the value in five frames to zoom in on the model.

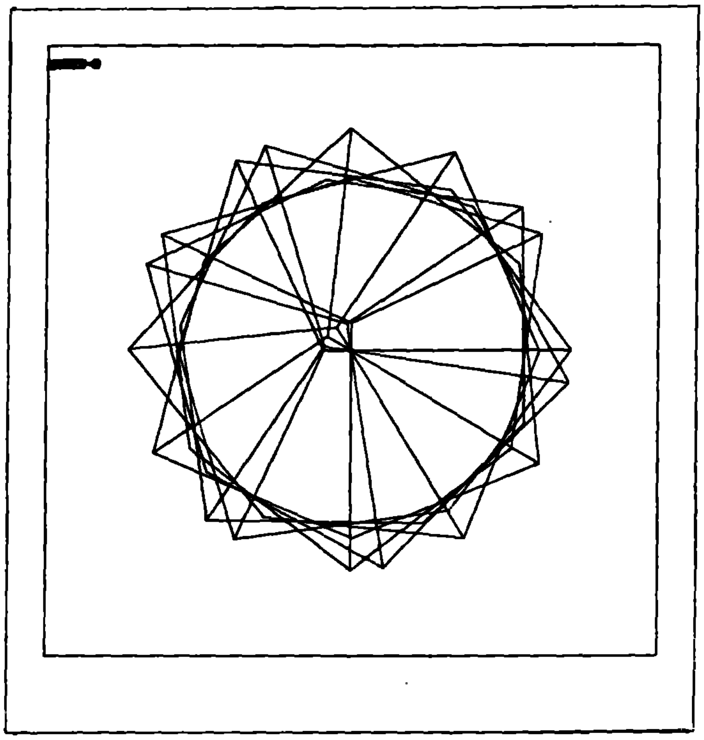


FIG. 41.

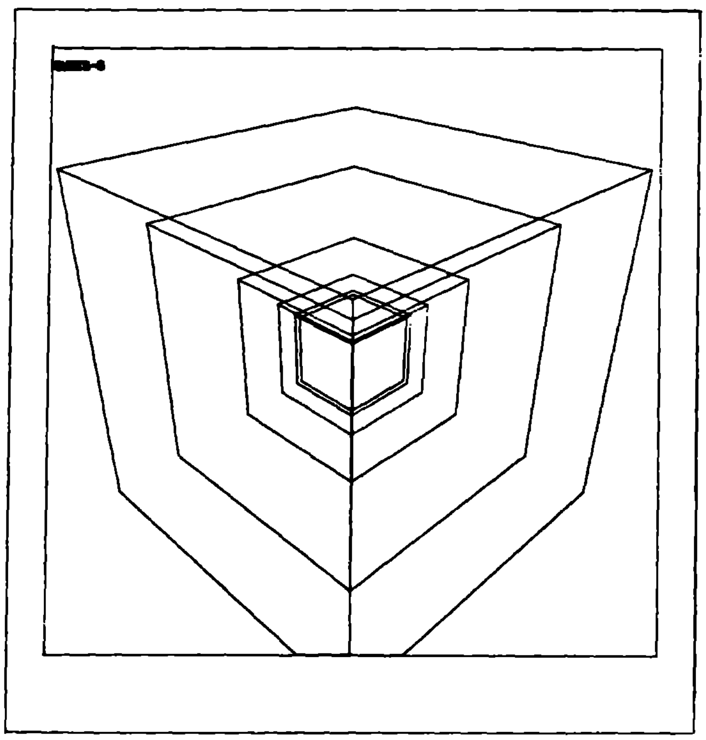


FIG. 42.

The PIVOT command for local rotations may also be animated, as well as EXPLODE and SLICE. These are all executed using the "elevator function" which has been demonstrated above.

COMMAND DEFINITIONS

Commands common to phases I and II

TV n	Use TMDS with monitor number n.
T or END	Terminate.
HEAD	Define heading to appear on all plots. Desired heading is expected on the next line in the input deck. If typed interactively, TAURUS will prompt for the heading.
C	Comment - proceed to next card.
TTY	This command may be used in the TAURUS command file to return control to the teletypewriter. When TAURUS finishes the last command in the command file, control is automatically returned to the teletypwriter if the last command was anything but T or END.
CFILE	This command may be used at the teletypewriter to return control to the command file specified on the execute line.

RJET n i

Send a copy of the FR8Ø file to rjet n using plot format i where

i=1 gives a 5" plot

i=2 gives a 8" plot

i=3 gives a 10.5" plot

i=4 gives the largest possible plot.

If i is negative, the plot is sideways, i.e., rotated 90 degrees clockwise on the paper. Plots may be sent to either the 11 or 22 inch plotters.

PLOTS

A complete record of the TMDS display starting with the next frame after this command is typed is written into an FR8Ø plotfile. If and only if the graphics file is not active, TAURUS will prompt for the box and ID when used interactively; otherwise, the box and ID will be read from the next line of the command file.

PHASE I COMMANDS

DAM	Display all materials (default).
M m	Display only material m.
P m	Display only part m.
DMS n m_1 $m_2 \ldots m_n$	Display material subset of n materials incuding materials m_1, $m_2, \ldots,$ and m_n.
DPS n p_1 $p_2 \ldots p_n$	Display part subset of n parts including parts p_1, $p_2 \ldots,$ and p_n.
CMN e_1 e_2 m	Change material number of elements e_1 to e_2 to m.
TIME t	State corresponding to time t is read into memory. If t does not correspond to a particular state in the database, TAURUS will interpolate between two states to time t. If t exceeds the maximum or minimum time in the database, TAURUS will extrapolate to time t using the last or first two states, respectively.
STATE n	State n is read into memory.
DSF s	Displacements are scaled by s. The default is 1.
DSFS s_x s_y s_z	Displacements in the x, y, and z directions are scaled by s_x, s_y, and s_z respectively.

XSCALE s	Scale x component of initial geometry by s.
YSCALE s	Scale y component of initial geometry by s.
ZSCALE s	Scale z component of initial geometry by s.
UDG n	Plot undeformed geometry as dashed lines if n=1 and as solid lines if n=-1. This command is turned off by setting n=0.
CONTOUR c	Contour component number c on all displayed materials. Component numbers are given in Table 1.
WV c w_x w_y w_z	Warp mesh by adding $v_c w_x$, $v_c w_y$ and $v_c w_z$ to the x, y, and z nodal coordinates, respectively, where v_c is the value of component c at the node. The warped mesh is then displayed. This command works best for planar models where the scale factors are a multiple of the direction cosine of the normal to the planar area. Component numbers are given in Table 1.
WC c w_x w_y w_z	Like the WV command above, but with contours also plotted over the warped geometry.
RANGE r_1 r_2	The contours range in value from r_1 to r_2. If not defined, TAURUS determines an optimal range. To turn off this command, repeat with r_1=r_2=0.
NUMCON n	n contours will be plotted. The default is set to 9.

Element Type	No.	Component
solid elements (2D & 3D)	1	x
	2	y
	3	z
	4	xy
	5	yz
	6	zx
	7	effective plastic strain
	8	pressure or average strain
	9	von Mises
	10	1st principal deviator maximum
	11	2nd principal deviator
	12	3rd principal deviator minimum
	13	maximum shear
	14	1st principal maximum
	15	2nd principal
	16	3rd principal minimum
hexahedrons membranes plates shells	17	x-displacement
	18	y-displacement
	19	z-displacement
	20	maximum displacement
	21	x-velocity
	22	y-velocity
	23	z-velocity
	24	maximum velocity
	25.	temperature (TACO3D)
membranes plates shells	26	M_{xx} bending resultant
	27	M_{yy} bending resultant
	28	M_{xy} bending resultant
	29	Q_{xx} shear resultant
	30	Q_{yy} shear resultant
	31	N_{xx} normal resultant
	32	N_{yy} normal resultant
	33	N_{xy} normal resultant
	34	surface stress $N_{xx}/t+6M_{xx}/t^2$
	35	surface stress $N_{xx}/t-6M_{xx}/t^2$
	36	surface stress $N_{yy}/t-6M_{yy}/t^2$
	37	surface stress $N_{yy}/t+6M_{yy}/t^2$
	38	surface stress $N_{xy}/t-6M_{xy}/t^2$
	39	surface stress $N_{xy}/t+6M_{xy}/t^2$
	40	effective upper surface stress
	41	effective lower surface stress
	42	maximum effective surface stress

TABLE 1. Component numbers for element variables. By adding 100, 200, 300, and 400 to component numbers 1 through 16 component numbers for infinitesimal strains, Green-St. Venant strains, Almansi strains, and strain rates are obtained, respectively.

RX θ	Rotate body θ degrees about the x axis in the screen coordinates. A positive rotation is counterclockwise.
RY θ	Rotate body θ degrees about the y axis in the screen coordinates. A positive rotation is counterclockwise.
RZ θ	Rotate body θ degrees about the z axis in the screen coordinates. A positive rotation is counterclockwise.
RESO r_h r_v	Horizontal and vertical resolutions are r_h and r_v, respectively. TAURUS defaults r_h=r_v=1024.
TRANS X Y Z	Shift the model's local origin to a new position (X, Y, Z).
XTRANS Δx	Translate picture a distance Δx in the screen coordinate system.
YTRANS Δy	Translate picture a distance Δy in the screen coordinate system.
ZTRANS Δz	Translate picture a distance Δz in the screen coordinate system.
CENTER	Center geometry.
WHERE	Print out the current transformation of the model and plot a picture similar to that shown in Fig. 3.

FOLLOW n	Node n will always be at the center of the display.
SUMM	Print out on the teletypewriter the minimum and maximum values of the coordinates and displacements. Also give the DIST, ANGLE, ZMIN, and ZMAX values.
ANGLE θ	The angle of view, θ degrees, is defaulted to 45. A smaller angle will reduce the perspective, and a larger angle will exaggerate the perspective.
ZMIN zmin	Reset front clipping plane to zmin.
ZMAX zmax	Reset back clipping plane to zmax.
SLICE zmin zmax	Slice model on a z-minimum plane, cap the sliced area as if it were bricks, and interpolate, if necessary, the scalar function onto the new surface for display.
DIST d	Distance between the origin of the model and the origin of the display space is set to d. Initially d is defaulted to 1.75 times the maximum of the X, Y, and Z ranges. Value, d, may be increased to shrink the model and decreased to enlarge or zoom in on the model.
TRIAD	Display a triad in the lower left-hand corner of the screen. The triad tracks the model coordinates as the model rotates. This option is turned off by the RESTORE command or by reissuing this command.

VIEW or V	Plot or display geometry with hidden lines removed.
DRAW or D	Plot or display geometry without hidden lines removed. This command is much cheaper that the VIEW command.
PVIEW or PV	Display each part with hidden lines removed. If the model has ten parts, then ten images will be displayed.
PDRAW or PD	As above, but without removing hidden lines.
NDPLT	Display mesh with nodal points labeled with node numbers.
EXPLODE n Δx Δy Δz p_1 $p_2 \ldots p_n$	Explode n parts including p_1, $p_2, \ldots,$ and p_n in the local coordinate direction $v = (\Delta x, \Delta y, \Delta z)$. Part p_1 is moved snv, part p_2 is moved s(n-1)v, part p_3 is moved s(n-2)v, and so on. The scale factor s defaults to 1 and is reset by the "ESF" command below. The "CENTER" command will center the exploded picture.
ESF s	Reset the scale factor for the vector components given in the "EXPLODE" command above to s.
PIVOT p_1 p_2 axis θ x_o y_o z_o	Rotate parts p_1 to p_n about the axis (specify x, y, or z) θ degrees. The relative origin (x_o, y_o, z_o) defines the point about which the rotations will take place. The model rotates about axes that are parallel to its original axis.

STICK	Display beam elements as lines rather than as long slender six sided solids (default). This command is valid if all elements are beam elements.
BEAM	Display beam elements as long slender six sided solids (default).
RESTORE	Zeros all translations and relative origins, destroys all reflected parts, initializes all rotation matrices to the identity matrix, resets scale factor to 1, and centers picture.
REVERSE	All polygons in the database have a clockwise orientation rather than the default counterclockwise orientation. If both orientations are present, the MIXED command may be invoked.
MIXED	Polygons in the database are not numbered consistently.
POOR	Any polygon whose outward normal faces away from the origin is neither drawn nor sent to the hidden surface algorithm. This is the default if bricks are the only elements in the database.
RAXY or ARXY	Reflect all parts about the xy plane. The geometry should be in its original orientation or in a rotated state of 180 degrees. The model and its reflected parts are automatically centered. If the displacement response is antisymmetric, the "ARXY" command should be used.

RAYZ or ARYZ	Reflect all parts about the yz symmetry plane. The geometry should be in its original orientation or in a rotated state of 180 degrees. The model and its reflected parts are automatically centered. If the displacement response is antisymmetric, the "ARYZ" command should be used.
RAZX or ARZX	Reflect all parts about the zx symmetry plane. The geometry should be in its original orientation or in a rotated state of 180 degrees. The model and its reflected parts are automatically centered. If the displacement response is antisymmetric, the "ARZX" command should be used.
R n m k	Repeat command. TAURUS will repeat the command that immediately follows for states n to m incrementing by k. This command applies to the "VIEW" and "CONTOUR" commands.
RMC n m k ℓ	Repeat multiple commands. TAURUS will repeat the commands that immediately follow, including the next ℓ lines of input for states n to m incrementing by k. This command is generally preferred over repetitious use of the I/O intensive "R" command.
MOVIE t_0 t_k k ℓ	Repeat multiple commands. TAURUS will repeat the commands that immediately follow, including the next ℓ lines of input, starting at time t_0 and ending at time t_k in k equal increments. TAURUS will extrapolate to times t_0 and t_k if necessary.

ANIMATE n (options, if any) ENDA	Create n frames with the next "VIEW", "DRAW", or "CONTOUR" command. Within the n frames, the model may be translated, rotated, vibrated, or exploded. These latter actions are controlled by the command options: TRNS, ROTA, VIBR, SFEX, and DELD given in Table 2. The command ENDA terminates the option input. If VIEW, DRAW, or CONTOUR are typed, the ENDA command is assumed.
PAUSE or STOP	A FORTRAN PAUSE is issued. Execution of TAURUS will continue after the linefeed command is given.

TRNS Δx Δy Δz	Translate model the distance (Δx, Δy, Δz) in n frames.
ROTA θ_x θ_y θ_z	Rotate model (θ_x, θ_y, θ_z) degrees in n frames.
VIBR m	Vibrate model m complete cycles in n frames.
SFEX Δs	Increase exploded view scale factor by Δs in n frames (see EXPLODE and ESF commands).
DELD Δd	Change the distance to the origin by Δd. A positive value zooms out, and a negative value zooms in.
PVT p_1 p_2 θ_x θ_y θ_z	Rotate parts p_1 to p_2, θ_x, θ_y, and θ_z degrees about the relative origin specified in the PIVOT command.

TABLE 2. Available options with the "ANIMATE" command.

TRPT	Print timing data for the VIEW and DRAW commands. This command is turned off by retyping TRPT or by the RESTORE command.
CSIZE n	Change size of plotted characters to n characters per line.
PHS2	Proceed to Phase II.

****All remaining commands in this section pertain to continuous tone color****

COLOR	Turns off TMDS and FR80 plots and turns on the DICOMED option. Material colors are set to their default values, the background is set to black,and the value of the diffused light in the image is set to 1/10. Also, all materials are set for fringing if the CONTOUR command is typed. This option is turned off by the "TV" command. A file is created that can be plotted on the DICOMED as explained in Appendix A.
CB red green blue	Reset color background. In Appendix B we correlate several colors with their red, green, and blue components.
CM m_1 m_n red green blue	Reset color of materials m_1 to m_n.
CP p_1 p_n red green blue	Reset color of parts p_1 to p_n.
CFM m_1 m_n i	Fringing is turned on if i=1 or off if i=0 materials m_1 to m_n.
CFP p_1 p_n i	Fringing is turned on if i=1 or off if i=0 in parts p_1 to p_n.

$SFC\ n\ r_1\ g_1\ b_1 \ldots r_n\ g_n\ b_n$

Define n fringes and set their red, r_i, green, g_i, and blue, b_i, components. TAURUS defaults n to 5, and the colors to blue, cyan, green, yellow, and red, respectively.

FLAT

Use flat element shading (default). The light source is at the tip of the truncated view volume. An angle is defined by the normal to the element and the line joining the light source and the point upon the element from which the normal emerges. The light intensity varies as the square of the cosine of this angle. The light intensity does not match at the element boundaries with this option.

SMOOTH

Smooth element shading (Gourand's shading) is invoked. The light intensity will match at element boundaries to produce a curved surface simulation.

UNIFORM

Element faces are shaded uniformly, i.e., the shading is constant over each individual element. The value used will be the average of the nodal values based upon the flat shading.

DIFM $m_1\ m_n\ d$

Ambient or diffuse light in material m_1 to m_n is set to d ($0 \le d \le 1$). The default value is 1/10.

DIFP $p_1\ p_n\ d$

Ambient or diffuse light in parts p_1 to p_n is set to d ($0 \le d \le 1$). The default value is 1/10.

PHASE II COMMANDS

PHS1	Return to Phase I.
ETYPE i	Element type i to be gathered with the ELEMENTS command below. Default is i=3 for DYNA3D, NIKE3D, and TACO3D and i=2 for GEMINI. Only one element type may be gathered. One, two, and three dimensional elements correspond to i=1,2, and 3 respectively. One-dimensional elements are not yet implemented for time histories.
ELEMENTS n e_1 e_2 ... e_n	Elements for time history plots include n elements of type i with numbers e_1, e_2,...e_n.
NODES n n_1 n_2 ... n_n	Nodes for time history plots include n nodes with numbers n_1, n_2,...,n_n.
MATLS n m_1 m_2 ... m_n	Materials for time history plots include n materials with numbers m_1, m_2,...,m_n.
GATHER	Causes TAURUS to read through the plot files and store the time histories for all the variables specified in the "ELEMENTS", "COMP", "NODES", "MATLS", and "IFNDS" commands. This command must be typed before any time histories can be plotted.
EXPDATA **filename**	Plot experimental data in file **filename** on next plot generated by any of the commands "NTIME", "ETIME",..., etc. which follow. The data structure of **filename** is described in Appendix D.

ETIME c n e_1 e_2 ... e_n	Plot component c for n elements with numbers $e_1, e_2, \ldots, e_n$. Component numbers 1 through 16 defined in Table 1 are applicable.
NTIME c n n_1 n_2 ... n_n	Plot component c for n nodes with numbers $n_1, n_2, \ldots, n_n$. Component numbers are defined in Table 2. The node numbers must be defined in the "NODES" command.
MTIME c n m_1 m_2 ... m_n	Plot component c for n materials with numbers $m_1, m_2, \ldots m_n$. Component numbers are defined in Table 4. The material numbers must be defined in the "MATLS" command.
NRTIME c n_1 n_2	Plot the difference in component c by subtracting the value of c at node n_2 from the value at node n_1. Component numbers are defined in Table 2. The node numbers must be defined in the "NODES" command.
GTIME c	Plot global variable c. Component numbers are defined in Table 5.
PRINT	Print plotted time history in file HSPBULL. Only data plotted after this command is printed.

No.	Component
1	x-displacement
2	y-displacement
3	z-displacement
4	x-velocity
5	y-velocity
6	z-velocity
7	x-acceleration
8	y-acceleration
9	z-acceleration
10	temperature (TACO3D)

TABLE 3. Component numbers for nodal time history plots.

No.	Component
1	x-rigid body displacement
2	y-rigid body displacement
3	z-rigid body displacement
4	x-rigid body velocity
5	y-rigid body velocity
6	z-rigid body velocity
7	x-rigid body acceleration
8	y-rigid body acceleration
9	z-rigid body acceleration

TABLE 4. Component numbers for material time history plots.

No.	Component
1	x-rigid body displacement
2	y-rigid body displacement
3	z-rigid body displacement
4	x-rigid body velocity
5	y-rigid body velocity
6	z-rigid body velocity
7	x-rigid body acceleration
8	y-rigid body acceleration
9	z-rigid body acceleration
10	kinetic energy

TABLE 5. Component numbers for global variable time history plots.

The next five commands apply only to plots generated by the ETIME, NTIME, NRTIME, MTIME, ITIME, FTIME, and GTIME commands.

ASET amin amax	Set minimum and maximum values on abcissa to amin and amax, respectively. If amin=amax=0.0 (default) TAURUS determines the minimum and maximum values.
OSET omin omax	Set minimum and maximum values on ordinate to omin and omax, respectively. If omin=omax=0.0 (default) TAURUS determines the minimum and maximum values.
ASCL f_a	Scale all abcissa data by f_a. The default is f_a=1.
OSCL f_o	Scale all ordinate data by f_o. The default is f_o=1.
SMOOTH n	Each data point is replaced by an average of 2n adjacent points. The default is n=0.

EXAMPLES

CYLINDER IMPACT ON RAIL

The cylinder impact calculation in the DYNA3D user's manual was chosen as the first example to demonstrate the usage of TAURUS. Figure 43 below lists the command file with the figure number of the resulting display indicated. If these same commands were typed interactively, TAURUS would print pertinent information on the user's teletypewriter; however, no additional information would be required.

Command	Resulting Figure
Box S14	
1625 V	44
RAXY RY 180 RAYZ V	45
TIME .0064	46
RY 60 V	47
RY 30 V	48
RX 15 V	49
CONTOUR 7	50
RJET 10 2 END	

FIG. 43. TAURUS input file for displaying the cylinder impact.

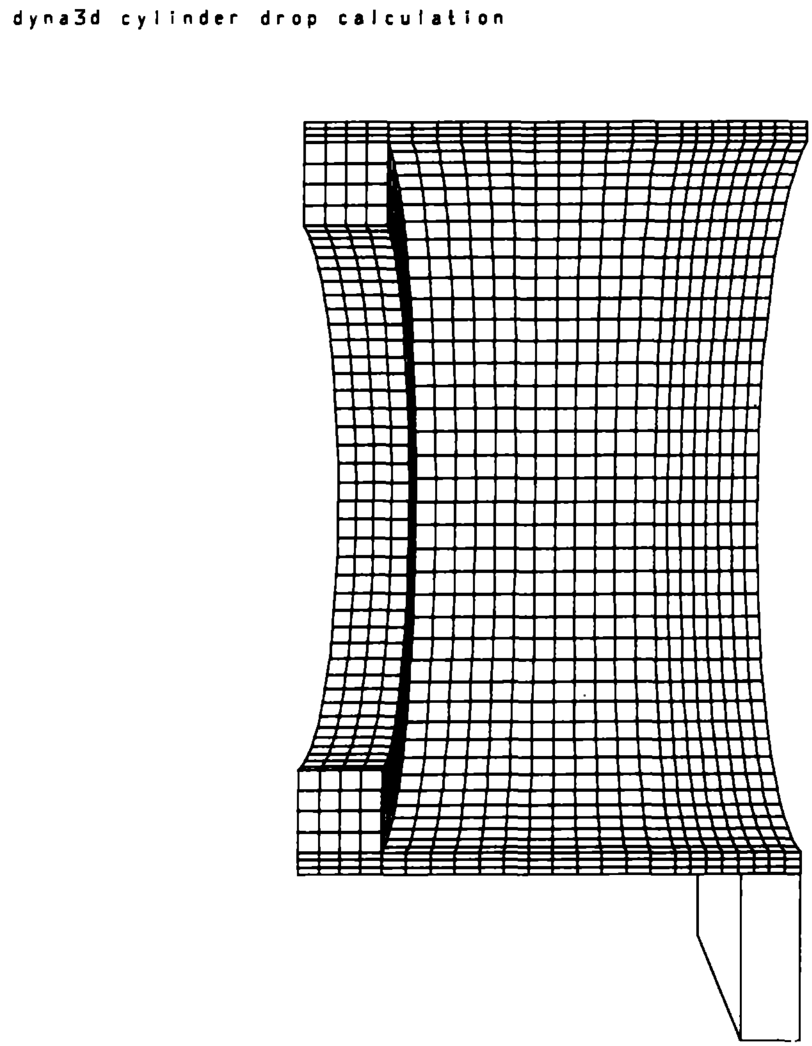


FIG. 44.

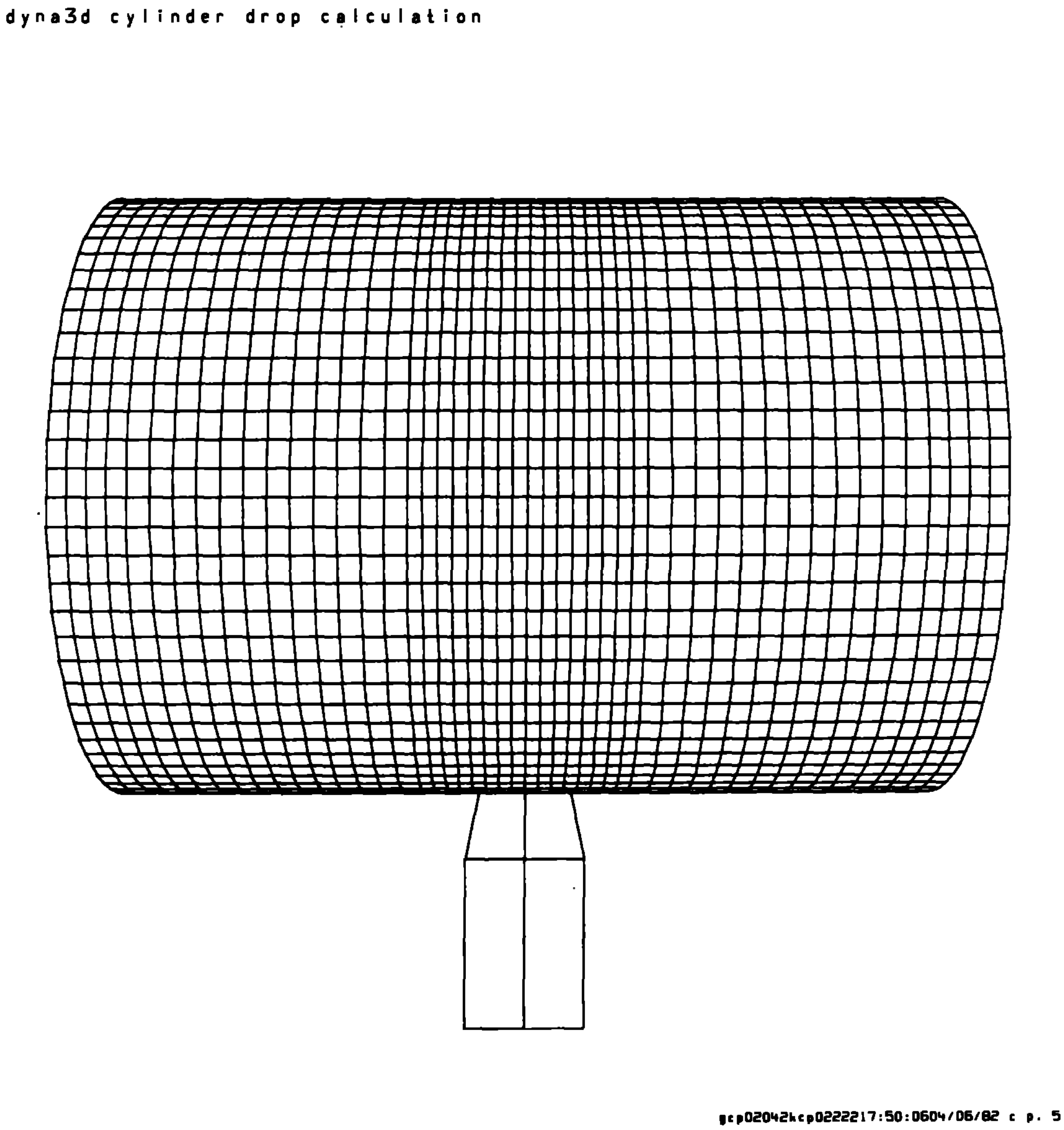

dyna3d cylinder drop calculation
gcp02042kcp0222217:50:0604/06/82 c p. 5

FIG. 45.

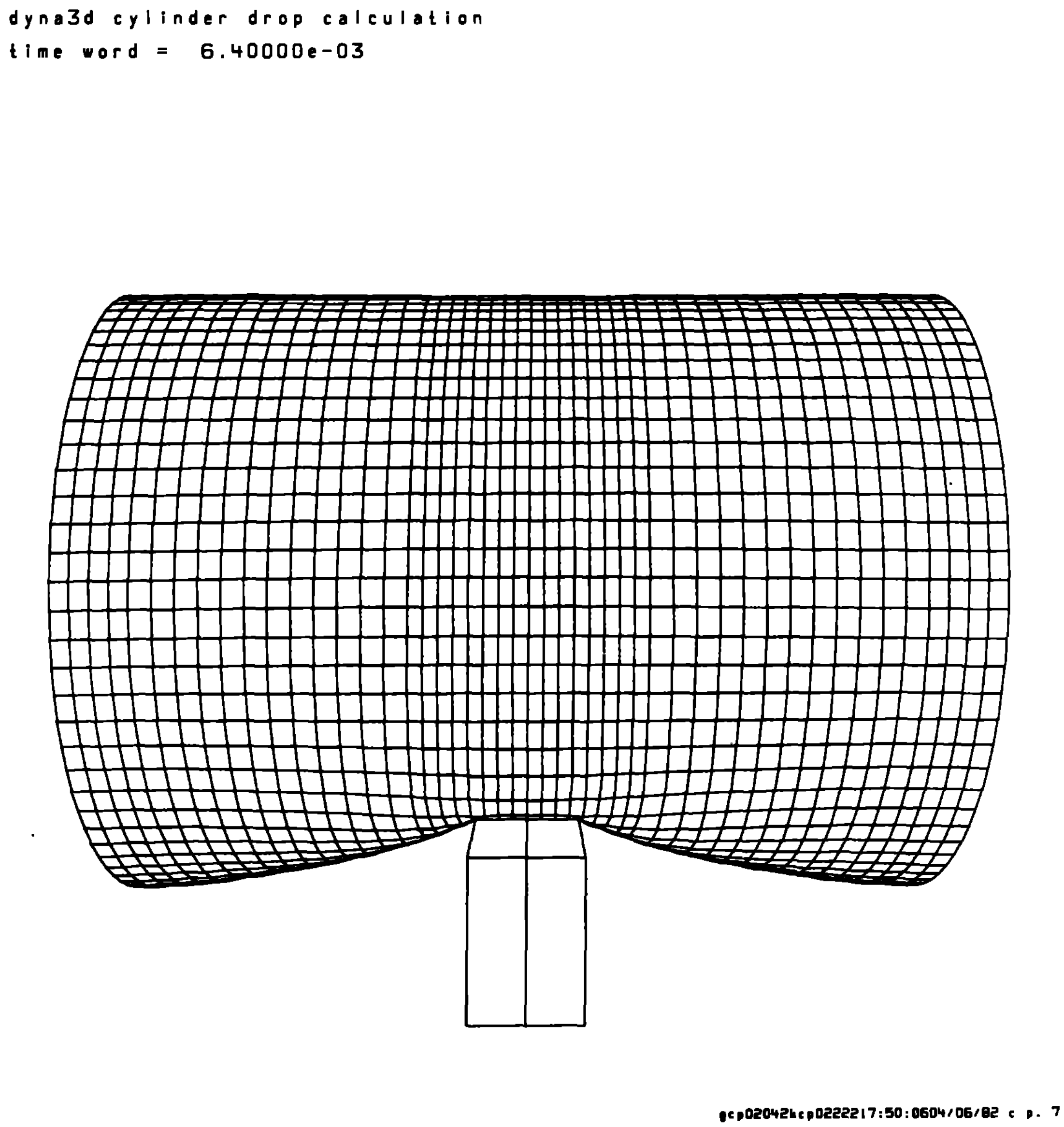
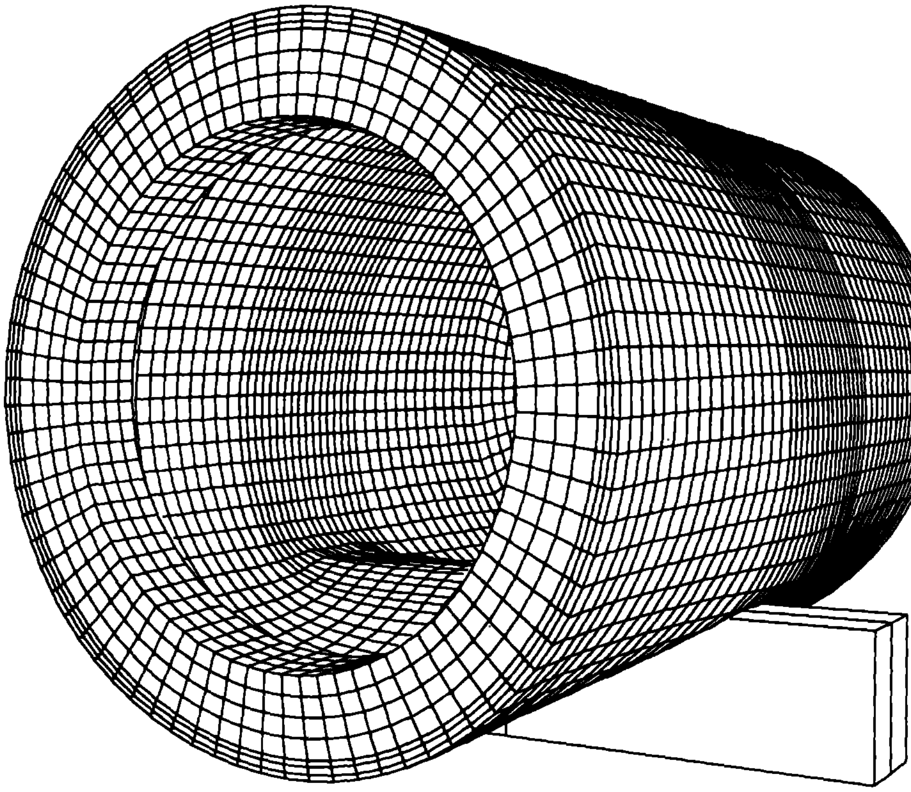

dyna3d cylinder drop calculation
time word = 6.40000e-03
gcp02042kcp0222217:50:0604/06/82 c p. 7

FIG. 46.

dyna3d cylinder drop calculation
time word = 6.40000e-03



gcp02042kcp0222217:50:0604/06/82 c p. 9

FIG. 47.

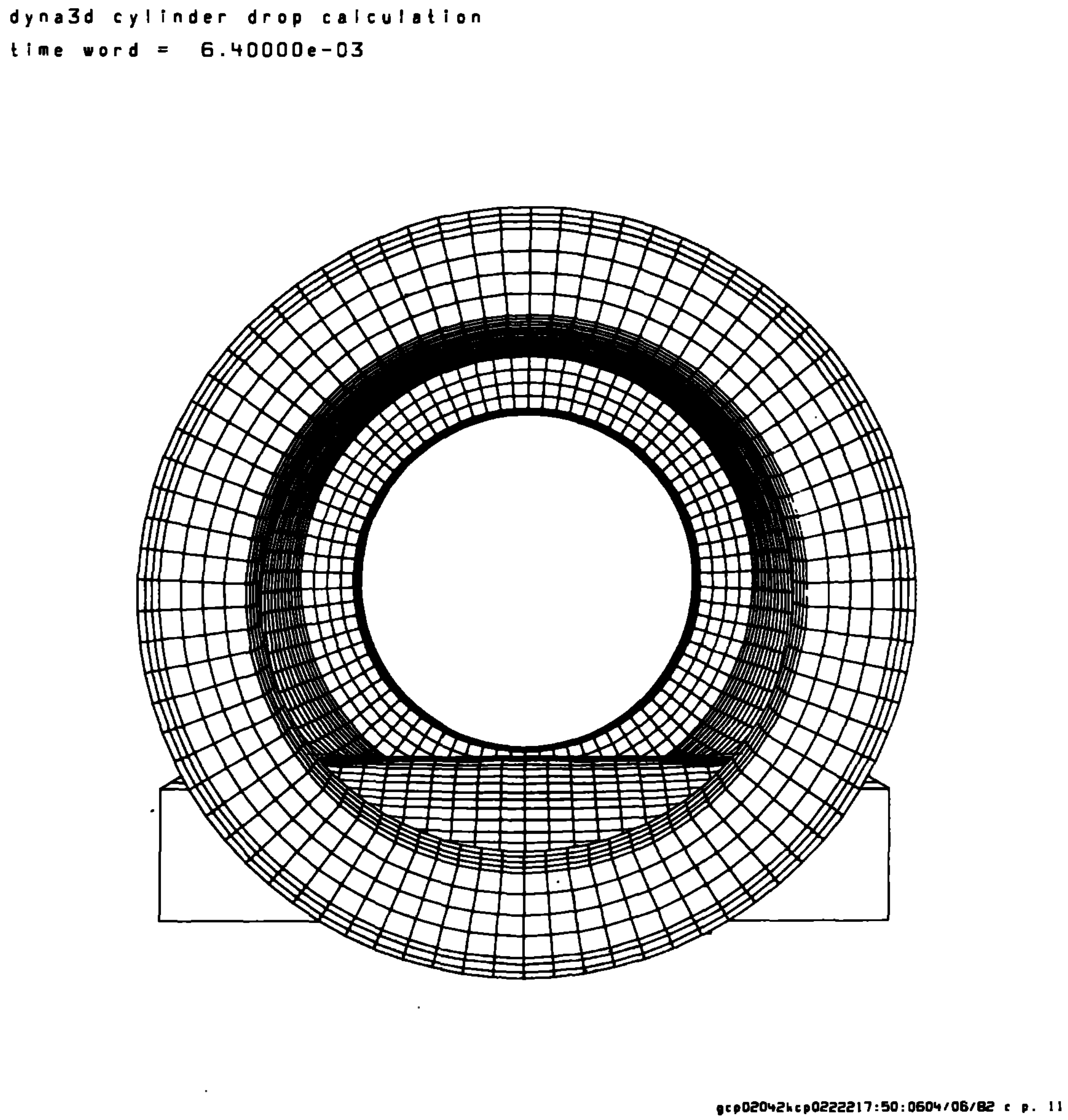

dyna3d cylinder drop calculation
time word = 6.40000e-03
gcp02042kcp0222217:50:0604/06/82 c p. 11

FIG. 48.

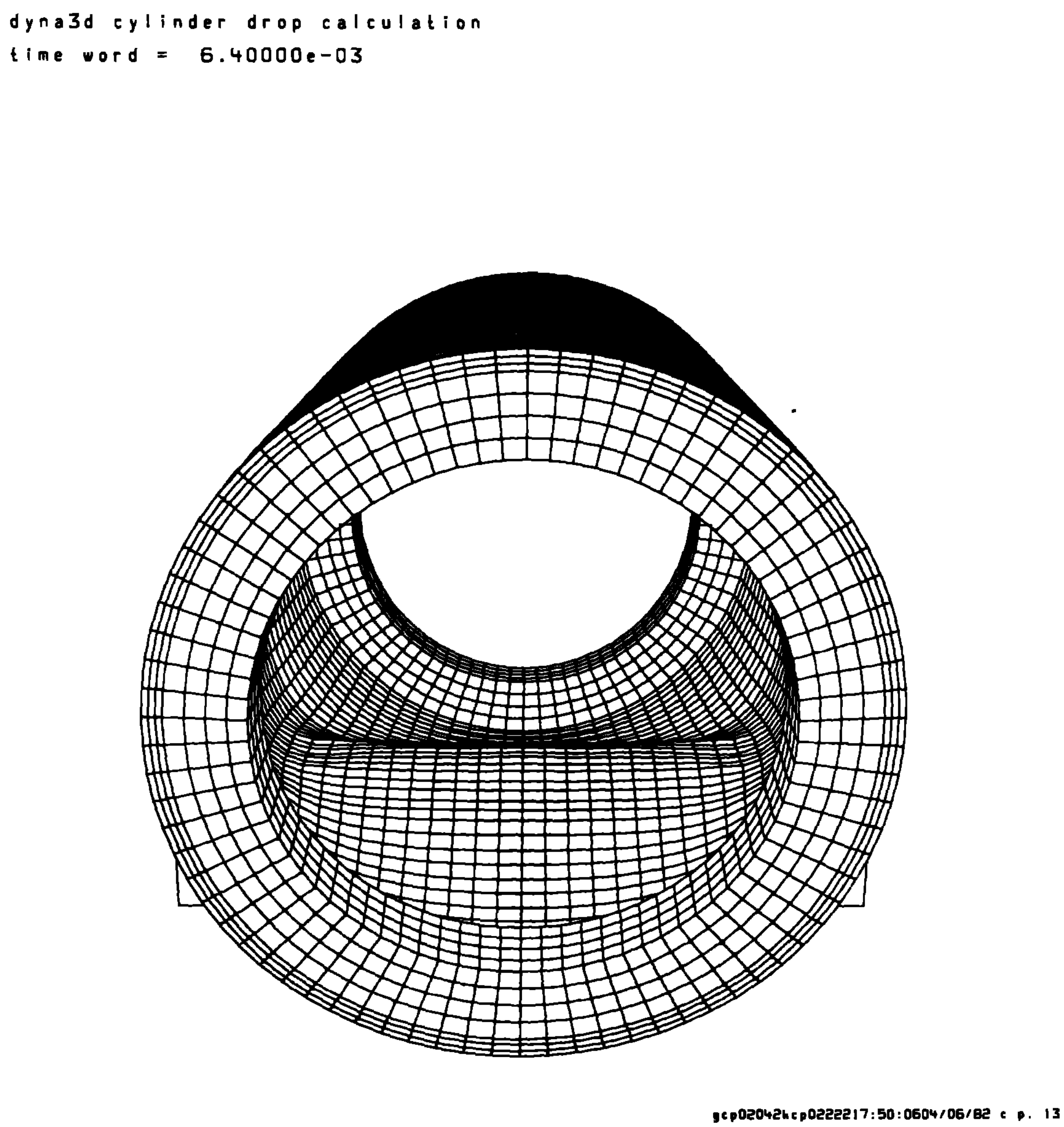

dyna3d cylinder drop calculation
time word = 6.40000e-03
gcp02042kcp0222217:50:0604/06/82 c p. 13

FIG. 49.

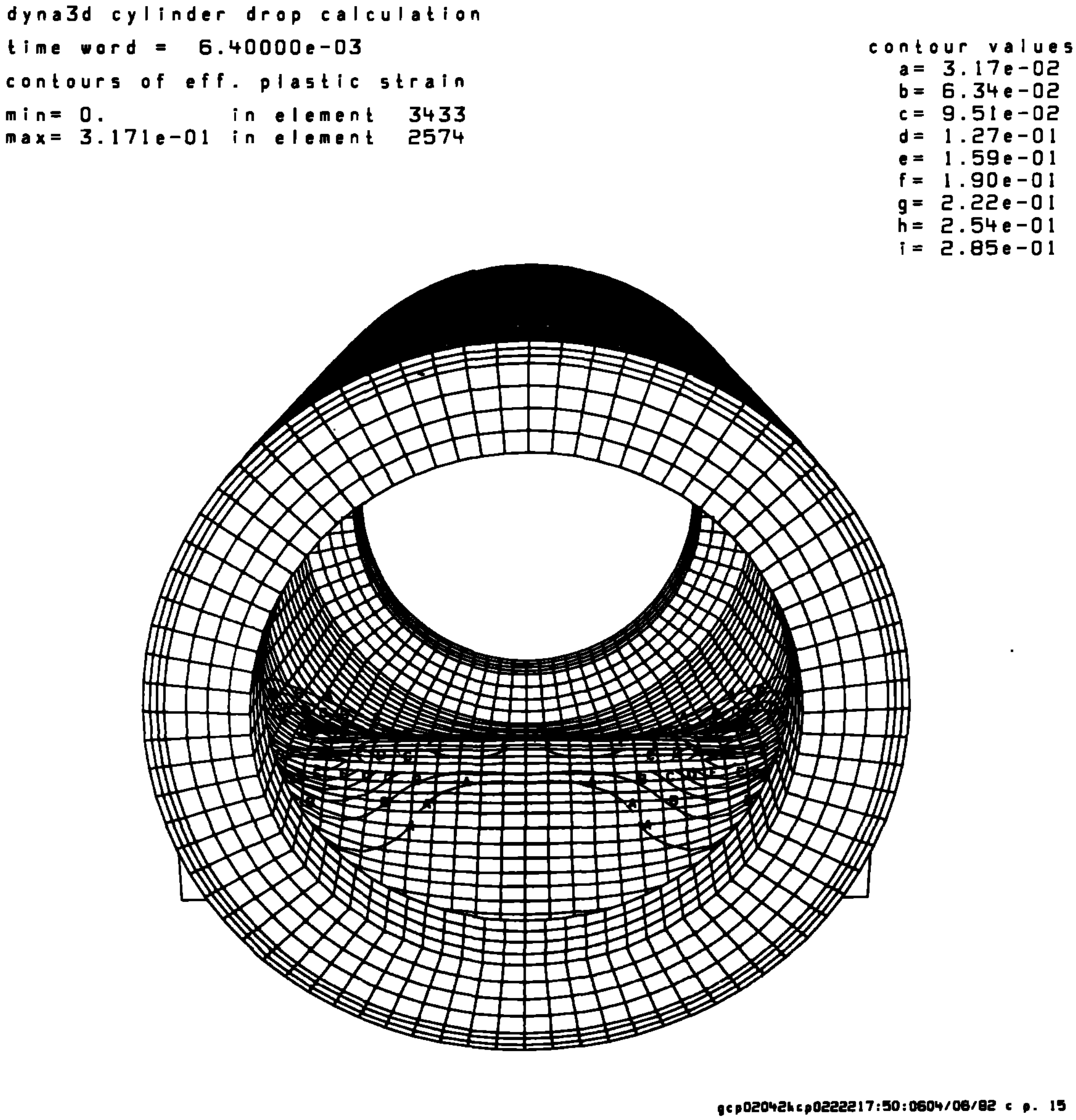

dyna3d cylinder drop calculation
time word = 6.40000e-03
contours of eff. plastic strain
min= 0. in element 3433
max= 3.171e-01 in element 2574
contour values
a= 3.17e-02
b= 6.34e-02
c= 9.51e-02
d= 1.27e-01
e= 1.59e-01
f= 1.90e-01
g= 2.22e-01
h= 2.54e-01
i= 2.85e-01
gcp02042kcp0222217:50:0604/06/82 c p. 15

FIG. 50.

BAR IMPACT ON RIGID WALL

The bar impact calculation in the DYNA3D user's manual is the second example. Figure 51 list the command file with the figure number of the resulting display indicated. An additional 9 color frames were generated but are not shown. A color file was generated, named "TSRL76, transported to the CDC7600 and written onto the tape for plotting on the DICOMED. Time history plotting is also demonstrated with this example.

Command	Resulting Figure
BOX S14	
1625	
HEAD	
DYNA3D BAR IMPACT ON RIGID WALL	
V	52
RAYZ RAZX RY -30 RX 15	
R 1 41 10 V	53 to 57
COLOR	
CB 0 1 0	
CM 1 2 1 1 1	
R 1 41 5 V	
TV 1625	
PHS2 ELEM 1 1 GATHER GTIME 6	58
GRID GTIME 6	59
NOGRID ETIME 7 1 1	60
RJET 15 2	
END	

FIG. 51. TAURUS input file for displaying the bar impact.

dyna3d bar impact on rigid wall

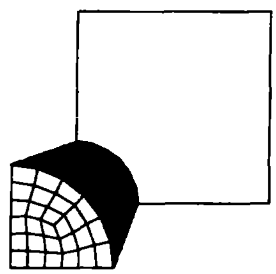


FIG. 52.

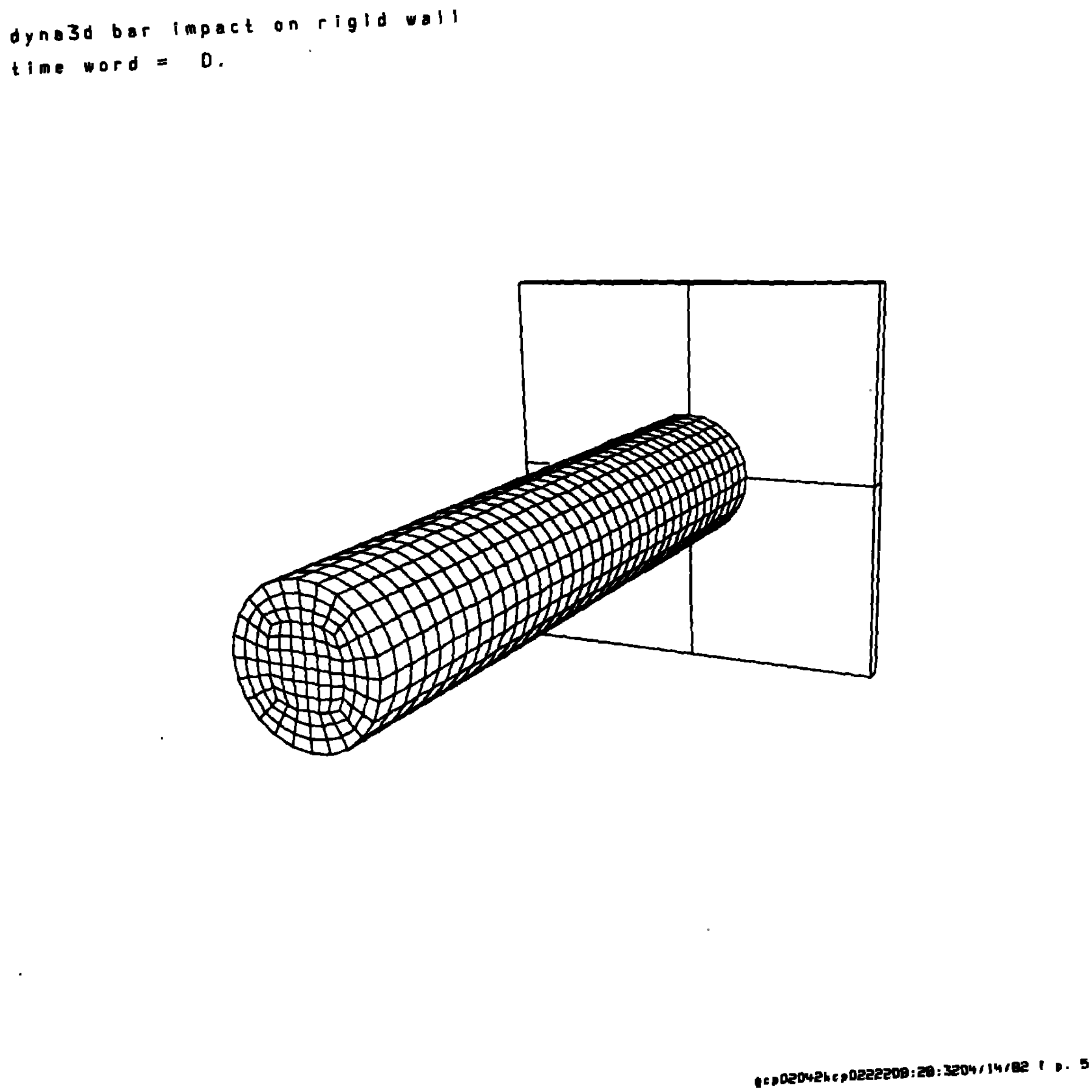
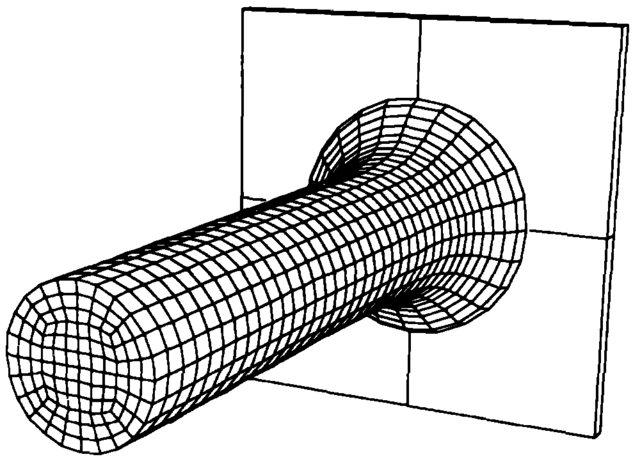

dyna3d bar impact on rigid wall
time word = 0.
gcp02042kcp0222208:20:3204/14/82 t p. 5

FIG. 53.

dyna3d bar impact on rigid wall
time word = 2.00097e+01



gcp02042bcp0222208:28:3204/14/82 f p. 7

FIG. 54.

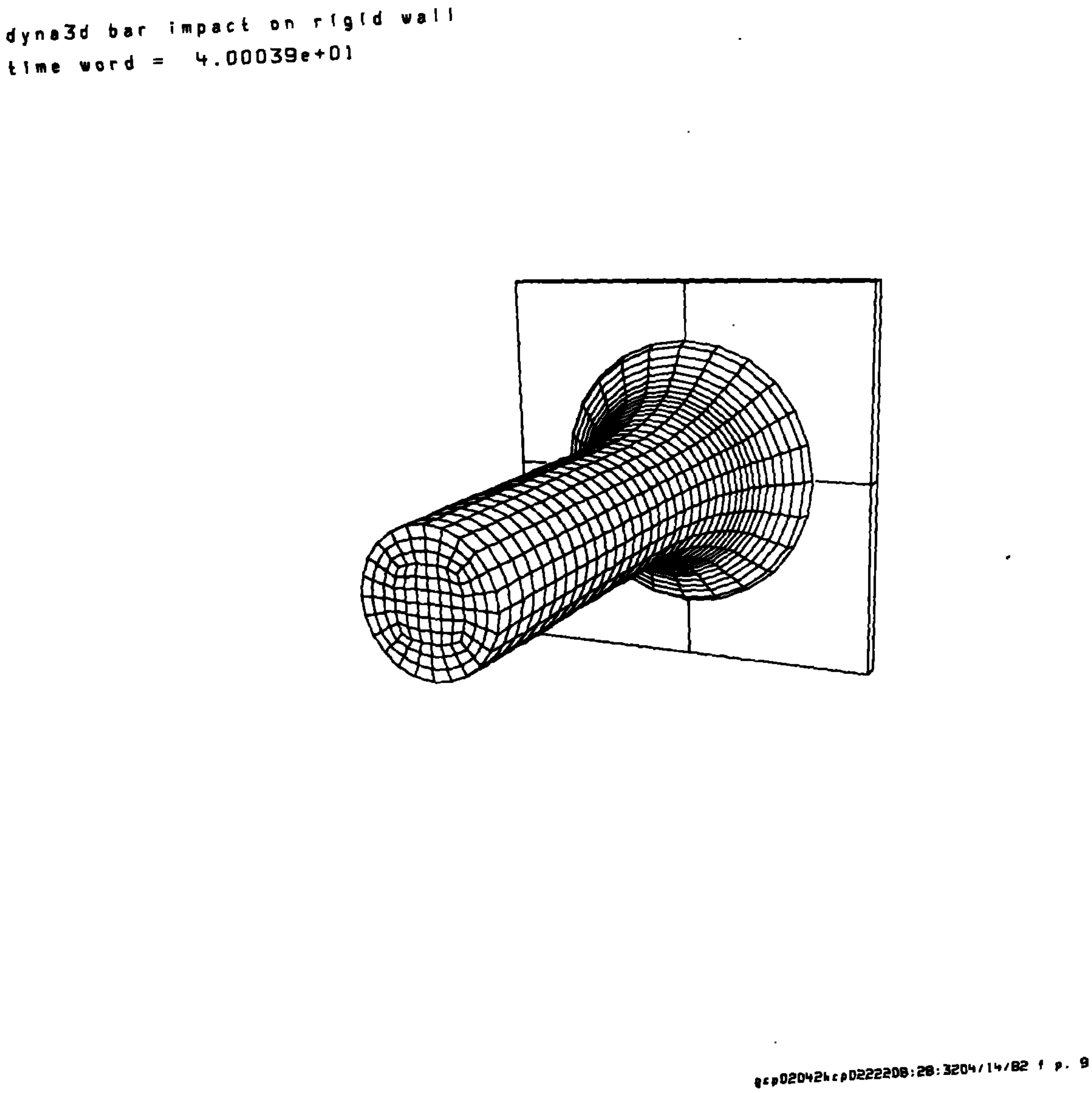
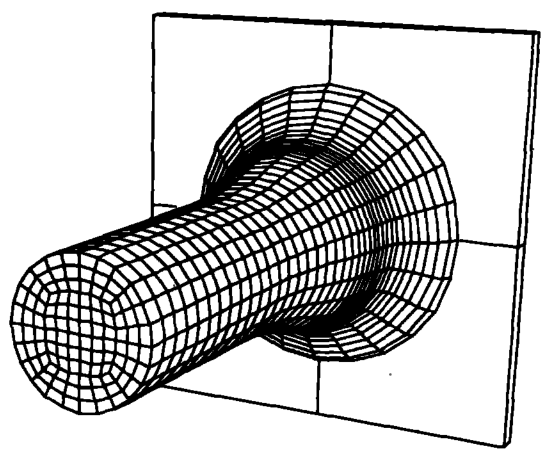

dyna3d bar impact on rigid wall
time word = 4.00039e+01
gcp02042hcp0222208:28:3204/14/82 f p. 9

FIG. 55.

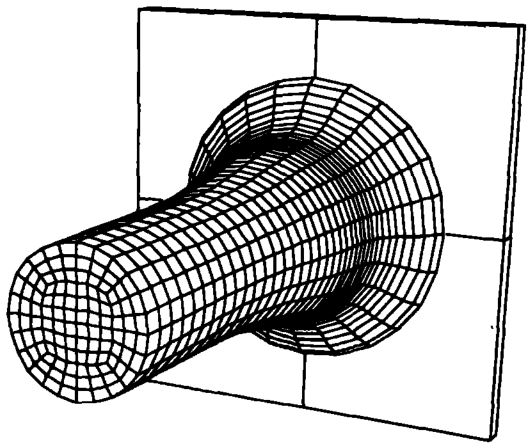
dyna3d bar impact on rigid wall
time word = 6.00023e+01



gcp02042kcp0222208:28:3204/14/82 f p. 11

FIG. 56.

dyna3d bar impact on rigid wall
time word = 8.00055e+01



gcp02042ktcp0222208:28:3204/14/82 f p. 13

FIG. 57.

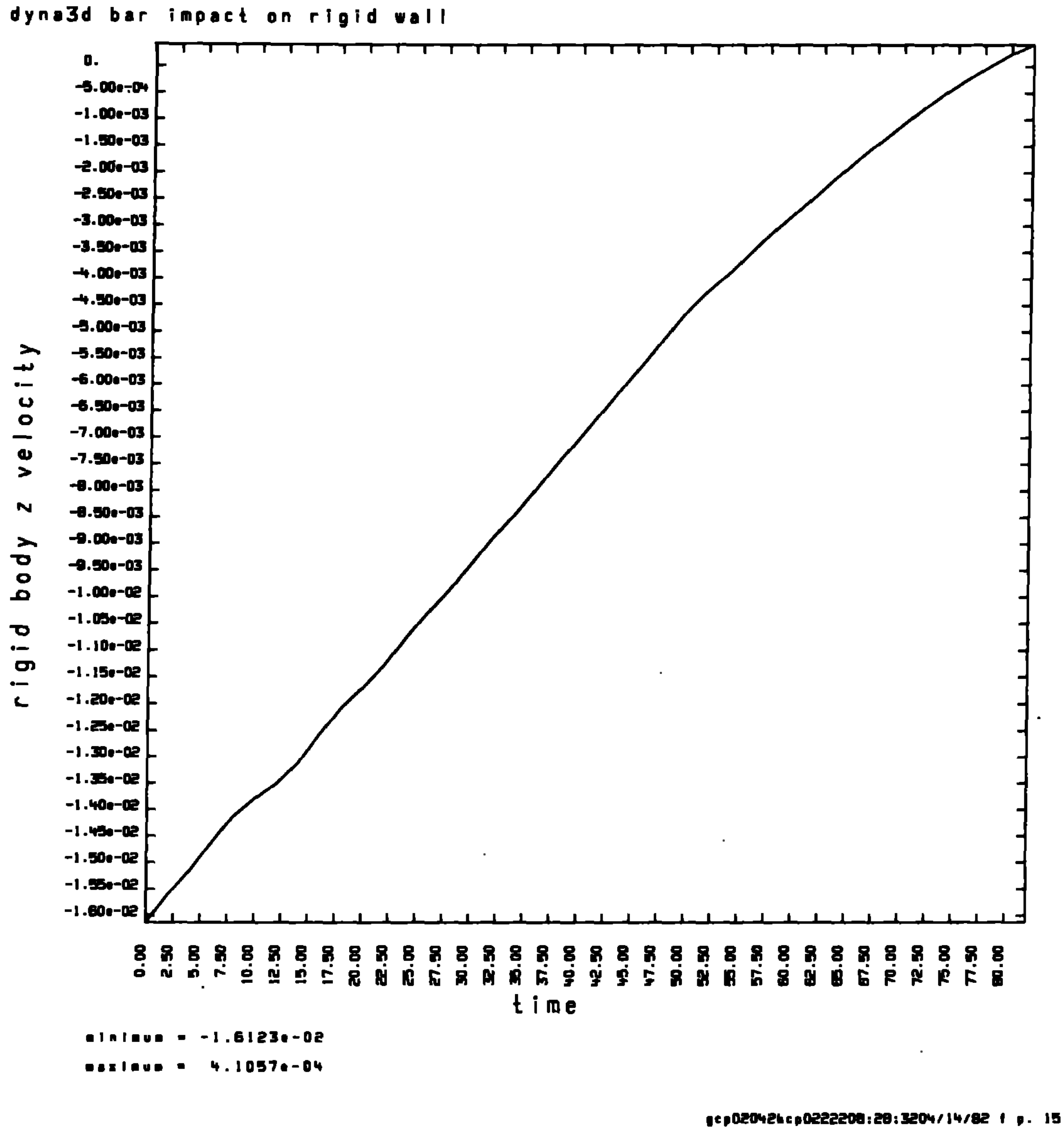

dyna3d bar impact on rigid wall
rigid body z velocity
time
minimum = -1.6123e-02
maximum = 4.1057e-04

FIG. 58.

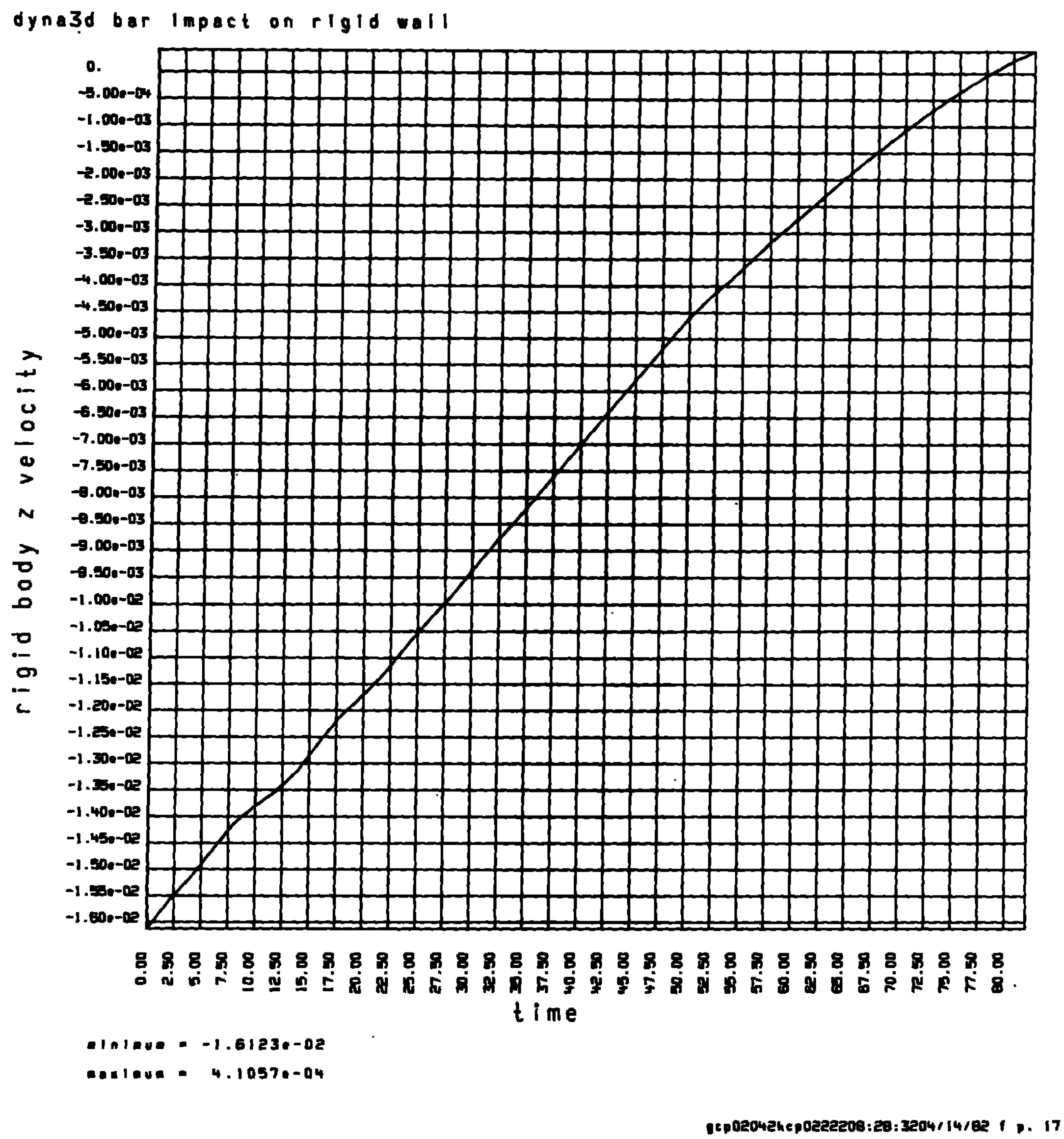

dyna3d bar impact on rigid wall
rigid body z velocity
time
minimum = -1.6123e-02
maximum = 4.1057e-04
gcp02042hcp02222008:28:3204/14/82 f p. 17

FIG. 59.

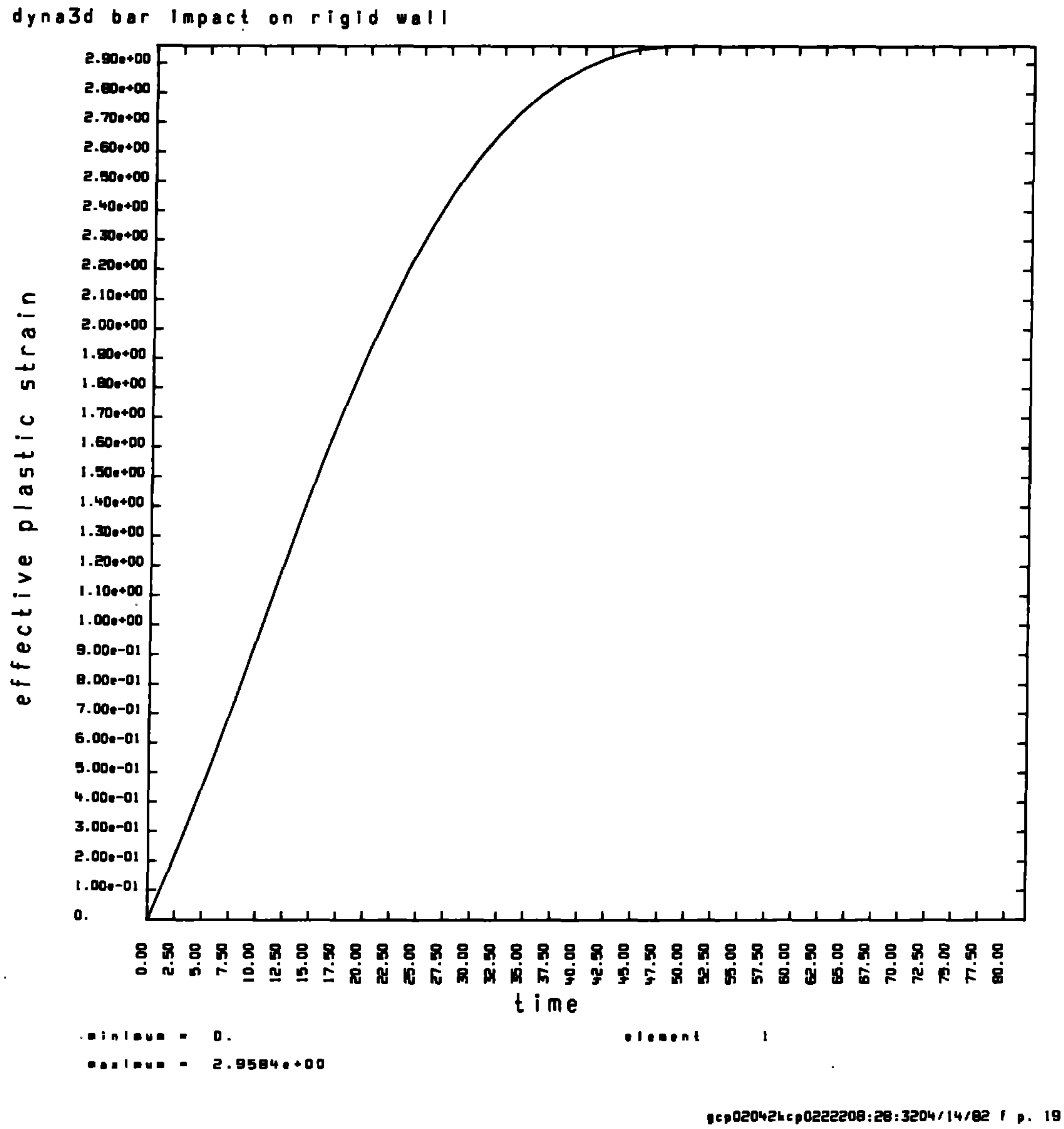

dyna3d bar impact on rigid wall
effective plastic strain
time
minimum = 0.
maximum = 2.9584e+00
element 1

FIG. 60.

LATERAL RAIL IMPACT

This last example, generated by R. A. Bailey for use in unclassified presentations on computer code capabilities at LLNL, shows a calculation of the lateral impact of a canister on a rail at 30 feet per second. Figure 61 list the command file with the figure number of the resulting display indicated. Use of the EXPLODE command is demonstrated in this file.

Command	Resulting Figure
Box S14	
306	
TRAN -1.5 0 0	
DIST 100	
RZ 90	
RY -90	
VIEW	62
EXPLODE 1 0 0 -30 1	
EXPLODE 1 0 0 -20 3	
EXPLODE 1 0 0 -10 4	
EXPLODE 1 0 0 -10 5	
EXPLODE 1 0 -10 5 2	
VIEW	63
RY -45	
VIEW	64
ANIMATE 3 SFEX -1 DELD -50 ENDA	
VIEW	65 - 67
MOVIE 0.000 0.012 6 1	
VIEW	68 - 74
RY 45	
XTRANS 4.5	
YTRANS 2.5	
DIST 15	
VIEW	75
END	

FIG. 61. TAURUS input file for displaying the cylinder impact.

lateral rail impact - 30 ft/sec

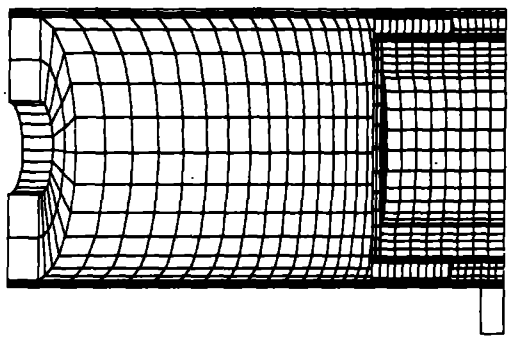


FIG. 62.

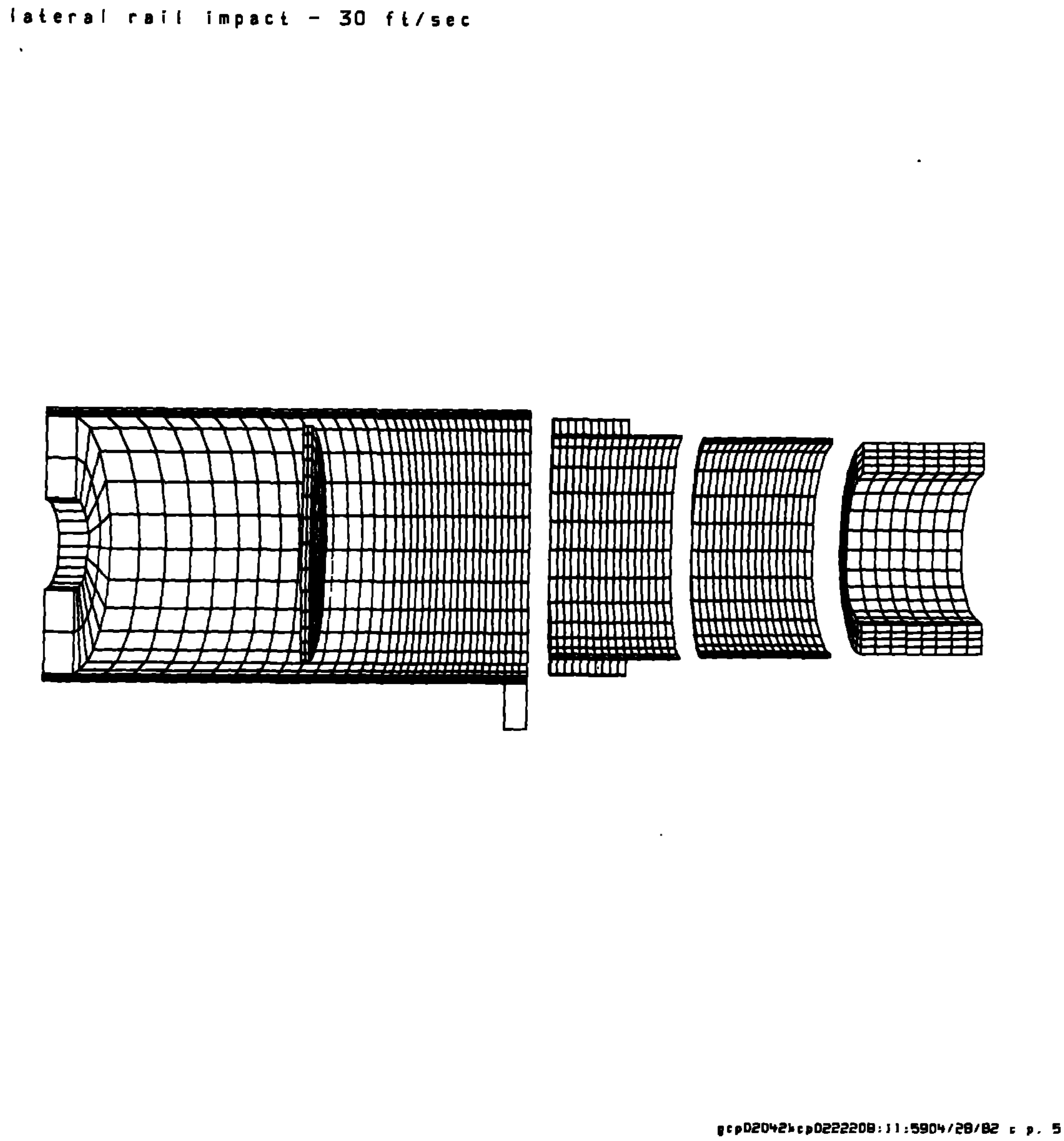

lateral rail impact - 30 ft/sec
gcp02042kcp0222208:11:5904/28/82 c p. 5

FIG. 63.

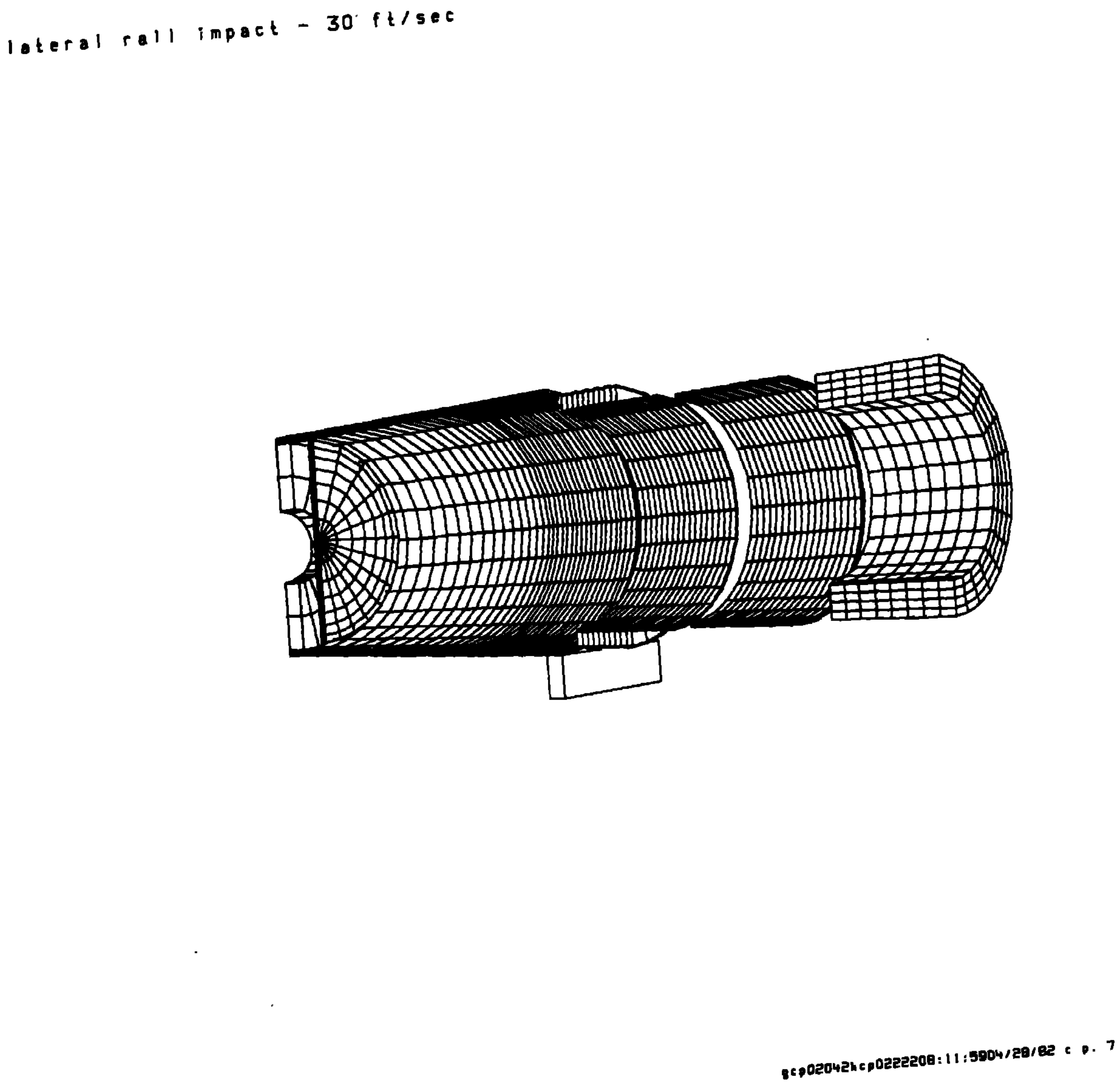
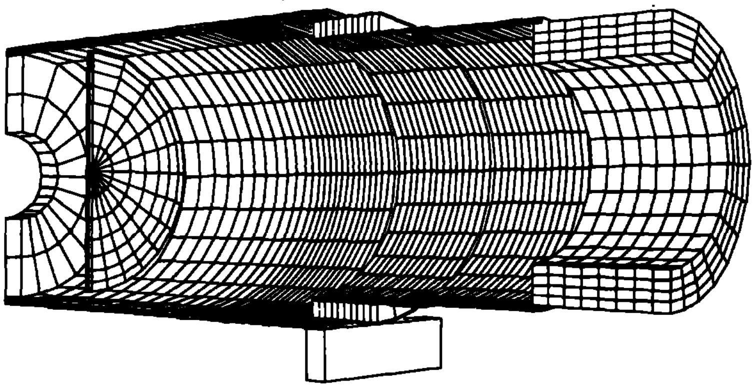

lateral rail impact - 30 ft/sec
gcp02042hcp0222208:11:5904/28/82 c p. 7

FIG. 64.

lateral rail impact - 30 ft/sec



gcp02042kcp0222208:11:5904/28/82 c p. 9

FIG. 65.

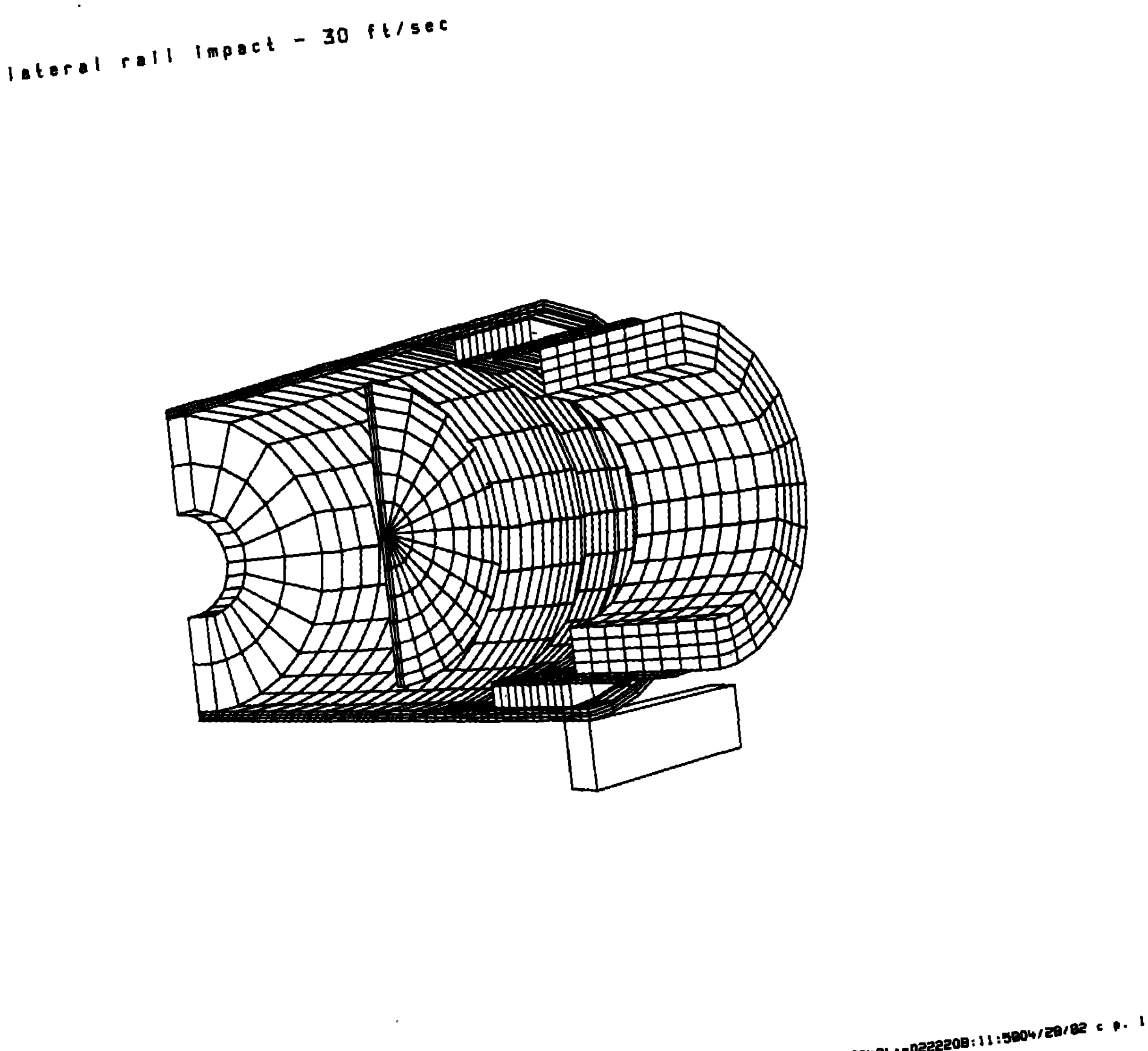

lateral rail impact - 30 ft/sec
gcp02042bcp0222208:11:5804/28/82 c p. 11

FIG. 66.

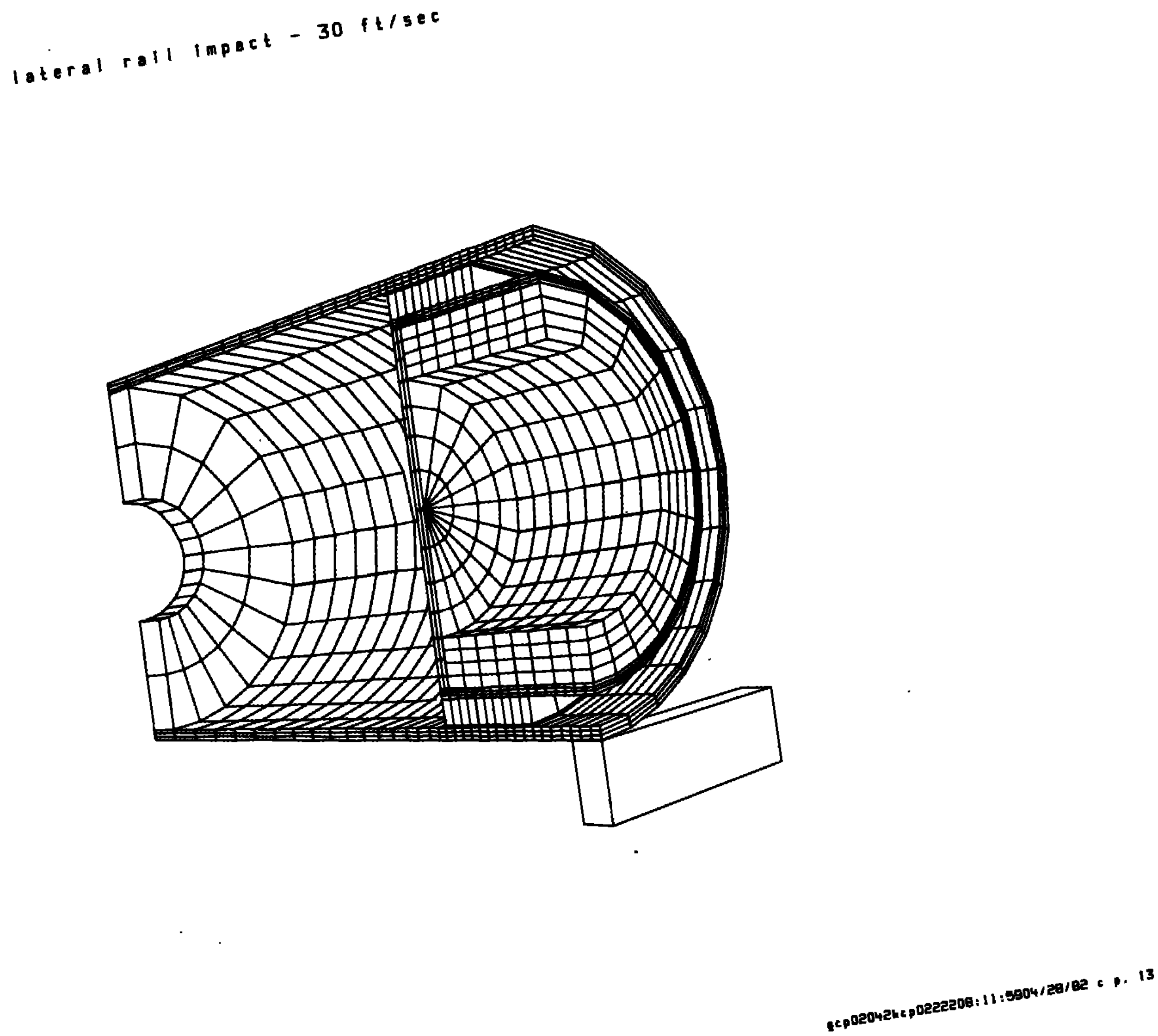
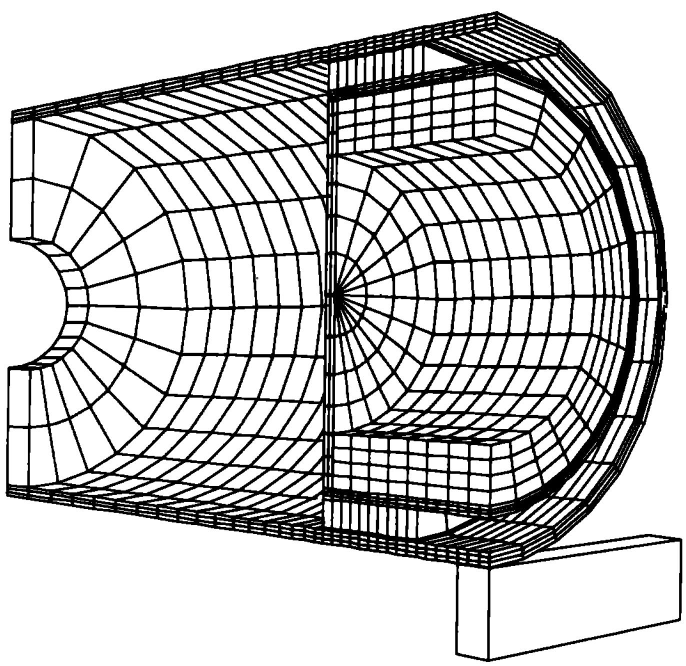

lateral rail impact - 30 ft/sec
gcp02042kcp0222208:11:5904/28/82 c p. 13

FIG. 67.

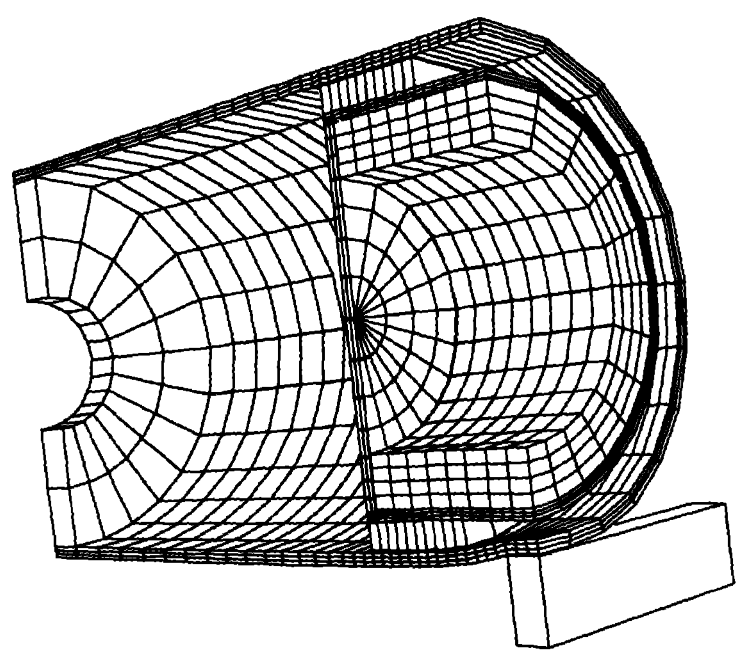
lateral rail impact - 30 ft/sec
time word = 0.



gcp02042kcp0222208:11:5904/28/82 c p. 15

FIG. 68.

lateral rail impact - 30 ft/sec
time word = 2.00000e-03



gcp02042kcp0222208:11:5904/28/82 c p. 17

FIG. 69.

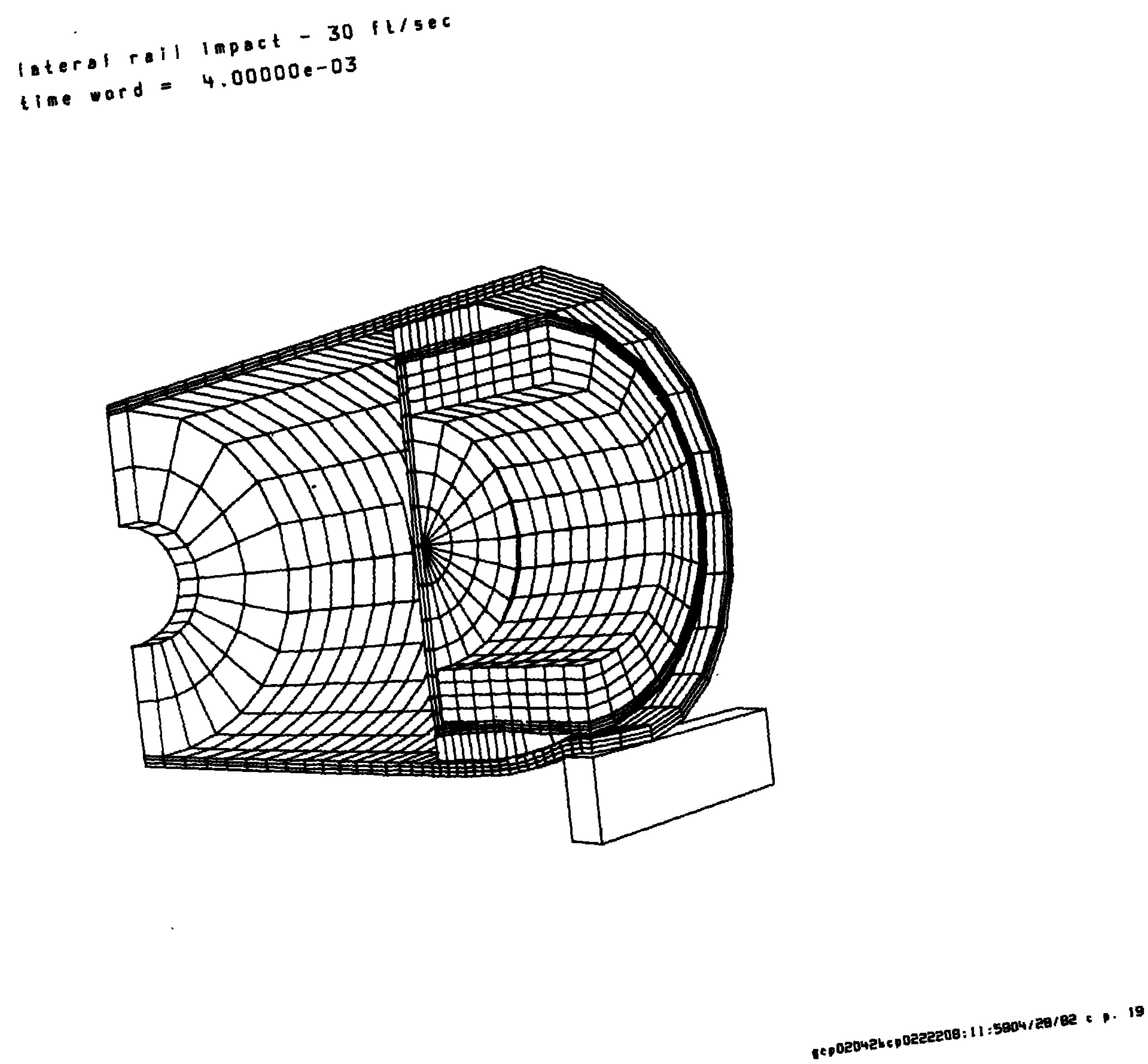
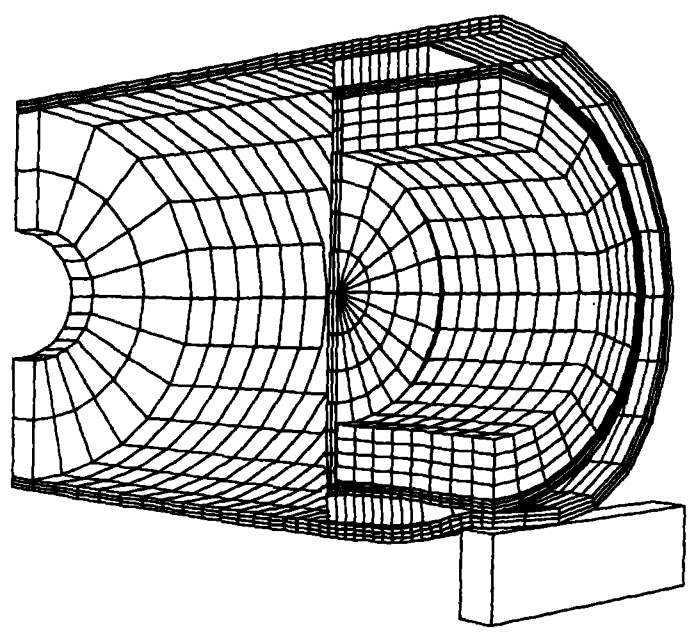

lateral rail impact - 30 ft/sec
time word = 4.00000e-03
gcp02042bcp0222208:11:5804/28/82 c p. 19

FIG. 70.

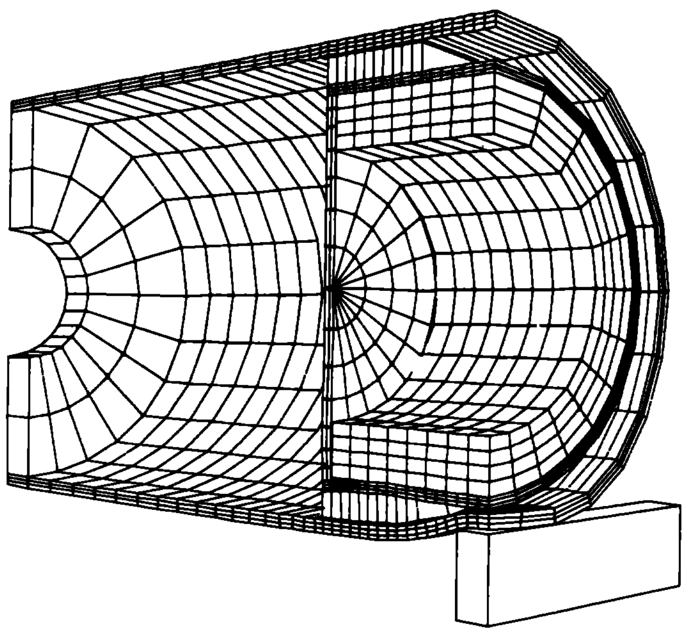
lateral rail impact - 30 ft/sec
time word = 6.00000e-03



gcp02042kcp0222208:11:5904/28/82 c p. 21

FIG. 71.

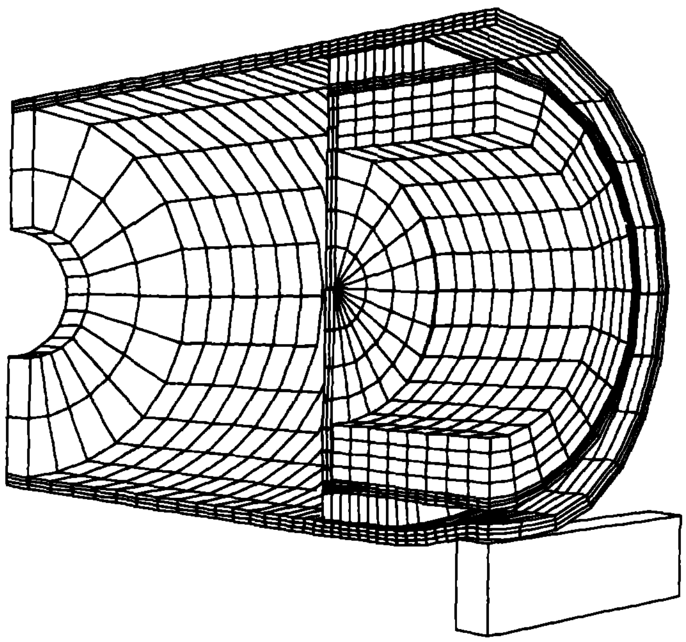
lateral rail impact - 30 ft/sec
time word = 8.00000e-03



gcp02042kcp0222208:11:5904/28/82 c p. 23

FIG. 72.

lateral rail impact - 30 ft/sec
time word = 1.00000e-02



gcp02042kcp0222208:11:5904/28/82 c p. 25

FIG. 73.

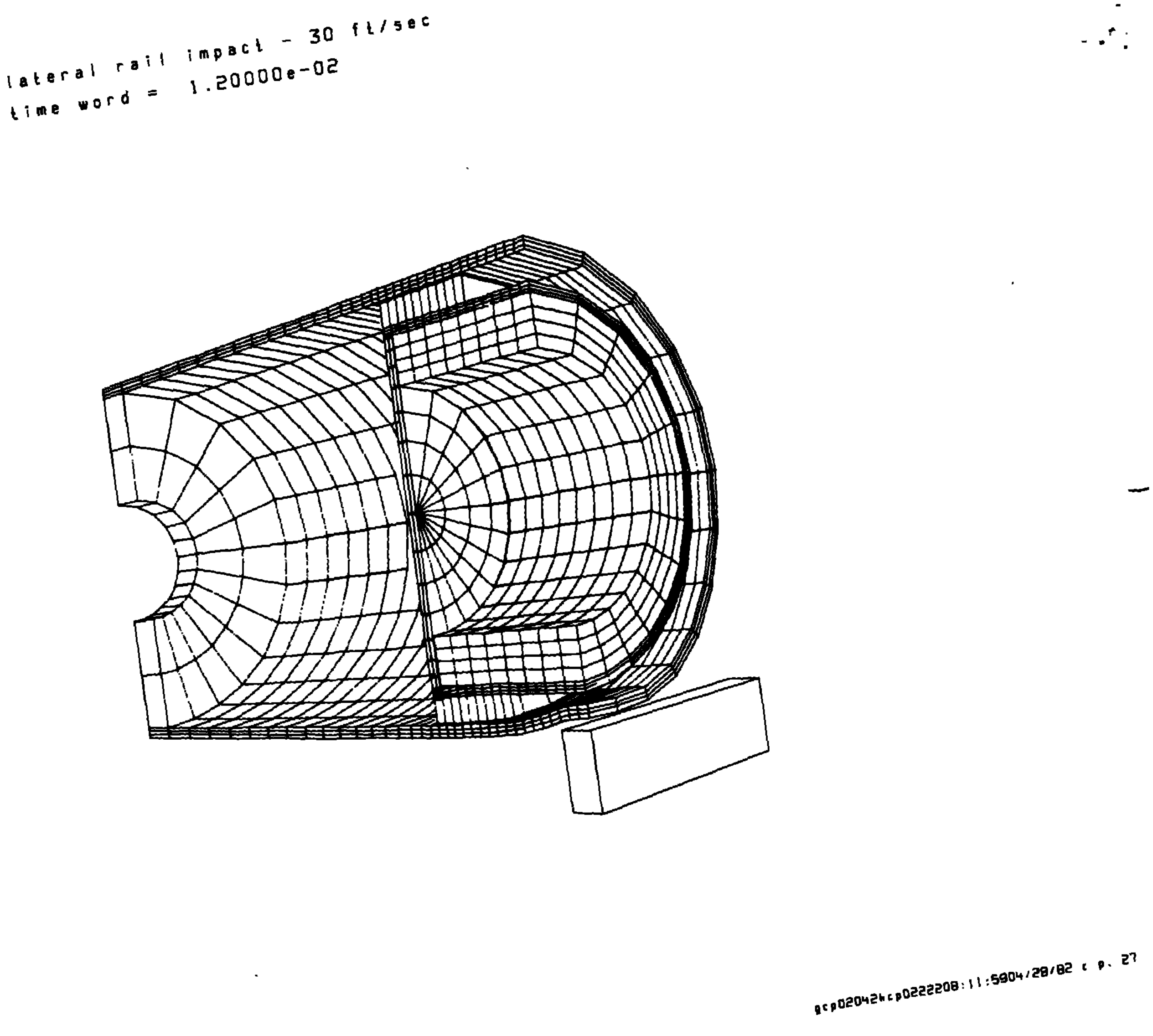

lateral rail impact - 30 ft/sec
time word = 1.20000e-02
gcp02042kcp0222208:11:5804/28/82 c p. 27

FIG. 74.

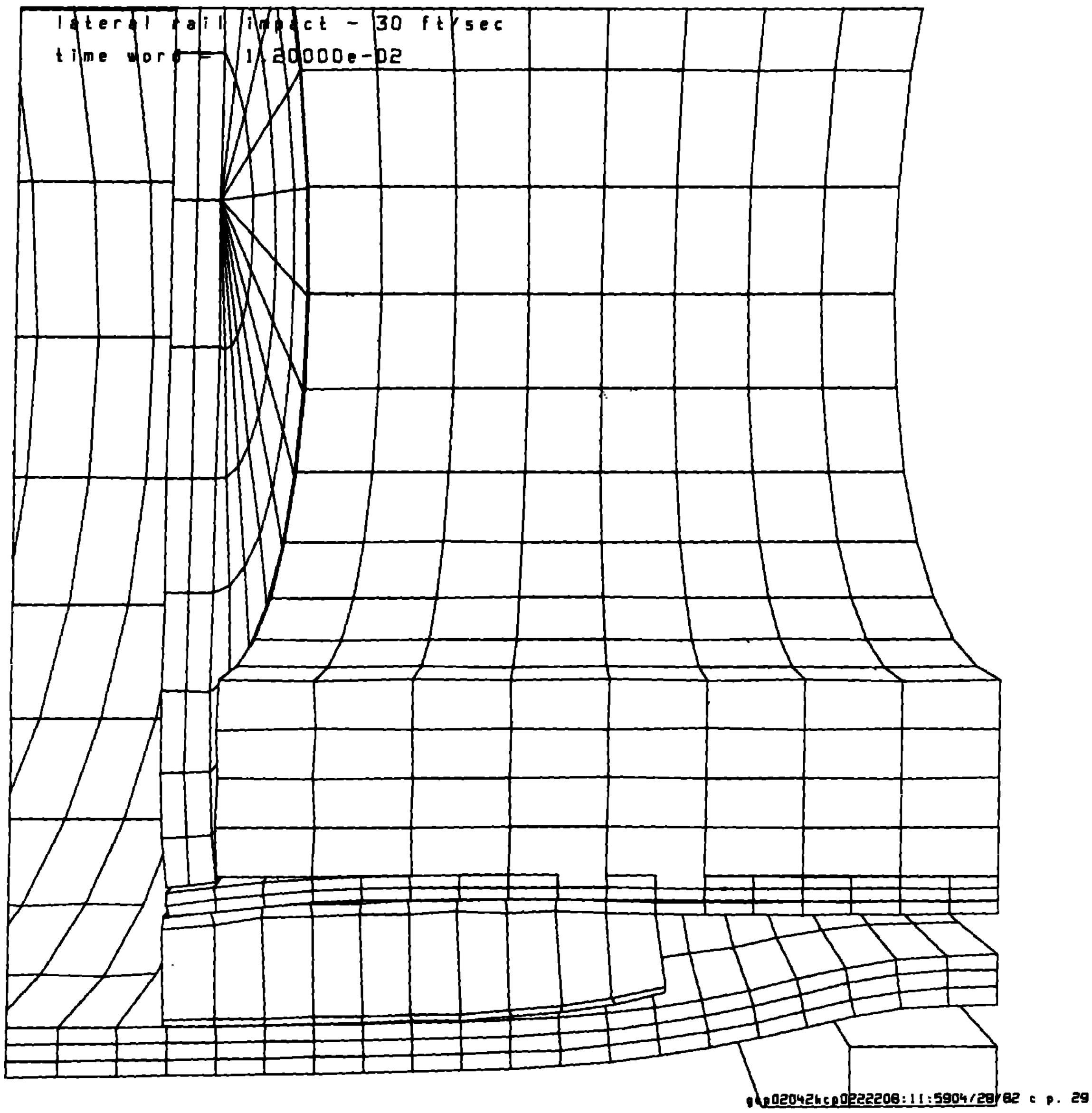

lateral rail impact - 30 ft/sec
time word = 1.20000e-02
0222208:11:5904/28/82 c p. 29

FIG. 75.

BINARY DATABASE

Unlike GRAPE, TAURUS can read only a binary database. We found with GRAPE that the MOVIE.BYU and SAMPP databases were not sufficiently used to justify their implementation into TAURUS.

There are three sections in this database. The first contains 64 words of control information. Only a portion of it is used presently and it allows some room for expansion. The second part of the database contains the geometry of the model. This is the nodal coordinates and the connectivity for the elements. A wide range of element types are allowed. The list includes: two-node beam and rod elements, three- and four-node membrane, plate, and shell type elements, and eight-node bricks. The third section contains the results of the analysis. The first data in this section is the state variables. The time word or other identification with any other quantities such as total energy of the state etc. is placed here. Then the nodal quantities such as temperature, displacement, velocity, and acceleration go into the file. Finally, the element quantities follow. These may be given for the whole element or for other points such as Gauss points within the element. Usually the stress tensor or stress resultants are given but other values as calculated may be used. The control words will indicate to TAURUS what is in the file. An indefinite number of states may follow the initial state.

CONTROL DATA

variable	#words	word #	description
TITLE	10	(1-10)	General title of the problem (8 characters per word.
RUN TIME	1	(11)	Time of day data generated.
DATE	1	(12)	Date of run.
MACH	1	(13)	Machine ran on.
CODE NAME	1	(14)	Code name of program generating data.
COMPILE DATE	1	(15)	Compile date for code or version number.
NDIM	1	(16)	Number of dimensions (2 or 3).
NUMNP	1	(17)	Number of nodal points.
ICODE	1	(18)	Flag for finite element program =0 no program (geometry only) =1 TACO =2 DYNA3D and NIKE3D =3 SAP4 (not maintained)

The following variables are only read if ICODE is 1, 2, 3, or 4.

variable	#words	word #	description
NGLBV	1	(19)	Number of global variables (to be read with each state).
IT	1	(20)	Flag for temperatures (=0 none, =1 read in the temperature for each node).
IU	1	(21)	Flag for displacements or current geometry (ICODE=2).
IV	1	(22)	Flag for velocities.
IA	1	(23)	Flag for accelerations.

These first twenty-three words are fixed. The following words vary according to the ICODE value.

Geometry only ICODE = 0

variable	#words	word #	description
NEL2	1	(24)	Number of 2 node elements
NUMAT2	1	(25)	Number of materials used in the 2 node elements.
NEL3	1	(26)	Number of 3 or 4 node elements
NUMAT3	1	(27)	Number of materials used in the 3 or 4 node elements.
NEL4	1	(28)	Number of 4 to 8 node elements (variable)
NUMAT4	1	(29)	Number of materials used in the 4 to 8 node elements.
NEL5	1	(30)	Number of 8 node brick elements
NUMAT5	1	(31)	Number of materials used in the 8 node bricks.
NEL6	1	(32)	Number of 8 to 20 node bricks (variable)
NUMAT6	1	(33)	Number of materials used in the 8 to 20 node elements.

TACO ICODE = 1 and NDIM = 2

variable	#words	word #	description
NEL4	1	(24)	Number of 4 node quads
NUMAT4	1	(25)	Number of material used in the 4 node quads

TACO ICODE = 1 and NDIM = 3

variable	#words	word #	description
NEL8	1	(24)	Number of 8 node bricks
NUMAT8	1	(25)	Number of materials used in the 8 node bricks.
NUMST	1	(28)	Number of states.

DYNA3D and NIKE3D ICODE = 2

variable	#words	word #	description
NEL8	1	(24)	Number of 8 node three-dimensional elements.
NUMAT8	1	(25)	Number of materials used in the 8 node three-dimensional elements.
NEL27	1	(26)	Number of 27 node three-dimensional elements (not yet implemented).
NUMAT27	1	(27)	Number of materials used in the 27 node three-dimensional element.
NVGP	1	(28)	Number of variables per three-dimensional element.
NEL2	1	(29)	Number of 2 node one-dimensional elements.
NUMAT2	1	(30)	Number of materials used in the 2 node one-dimensional elements.
NV1D	1	(31)	Number of variables per 1D elements.
NEL4	1	(32)	Number of four node two-dimensional elements, i.e., plate and shell elements.
NUMAT4	1	(33)	Number of materials used in the 4 node two-dimensional elements.
NV2D	1	(34)	Number of variables per two-dimensional elements.

Material numbers of elements in the database should range between 1 and NUMMAT inclusive when NUMMAT = NUMAT8 + NUMAT27 + NUMAT2 + NUMAT4. Material numbers for each element class should be unique to that class to ensure that TAURUS part numbers correspond to material numbers.

SAP4 ICODE = 3

variable	#words	word #	description
NUMEG	1	(24)	Number of different element groups used by SAP4.
NEIGEN	1	(25)	Number of eigenvalues.
NELTYP	NUMEG		Array of element type numbers for each group. In the lower six bits, and the number of variables per element for element data in the upper bits.
NELGRP	NUMEG		Array of number of elements per group.
NMATGP	NUMEG		Array of number of materials per group.

This is the end of the control section, the first 64 words of the file.

GEOMETRY DATA

The geometry section contains the nodal coordinates and the element connectivity. Node numbers are not stored in the database so interpolating nodal coordinates is not an option of this code. The order of the nodes is assumed to be the same as the order of the nodal data in the state (i.e. the temperatures at each node are in the same order as the original nodal coordinates). The connectivity is assumed to be three integers per 60 bit (CDC 7600) word, the exceptions to this is the ICODE=0 database. The order of the elements are 3, 2, and 1 dimensional elements if the ICODE=2 database is used.

variable	length	description
X	NDIM*NUMNP	Array of nodal coordinates. for 3-D codes X1,Y1,Z1,X2,Y2, Z2,... Xn,Yn,Zn. for 2-D codes X1,Y1,X2,Y2,...Xn,Yn. or R1,Z1,R2,Z2,...Rn,Zn.
IP	LENIP	Element array where LENIP is its length as determined by control section. The order of the data should be nodes defining a single element, followed by the material number for that element.

STATE DATA

The state data has three parts. The first is the time word and array of variables for the state (such as total energy, etc.). The second section contains the nodal data. This data may include temperature, displacement, velocity and accelerations. The third section contains the element data - the stress tensor or other variables for the element or the Gauss points within the element. TAURUS processes this data and extrapolates the values to the nodes for display.

Control Section

variable	length	description
TIME	1	Time word.
GBLV	NGLBV	Global variables for this state.

Nodal Section

variable	length	description
AN	NND	Total nodal variables for state, where NND = (IT + NDIM* (IU + IV + IA))*NUMNP or NND = NDIM*NUMNP for SAP4 eigen vectors. Deformed geometry, not displacements, is expected if ICODE=2.

Element Section

variable	length	description
EN	ENN	Total element data for state where ENN = NEL8*NVGP for ICODE = 2.

This state section is then repeated for each state in database.

Element data is data given at the integration points within the element. The contour plotter for line drawings and the color fringing algorithm require that the data be extrapolated to the nodal points. For centroidal data, the value is assigned to each of the eight nodes. The values at the nodes used by more than one element are then averaged.

Currently the values in the DYNA3D/NIKE3D database for hexahedron elements are:

1. Sigma-XX Cauchy Stress
2. Sigma-YY Cauchy Stress
3. Sigma-ZZ Cauchy Stress
4. Sigma-XY Cauchy Stress
5. Sigma-YZ Cauchy Stress
6. Sigma-ZX Cauchy Stress
7. Effective Plastic Strain or another variable depending on material.

For membrane, plate and shell elements the values are:

1. M_{xx} bending moment
2. M_{yy} bending moment
3. M_{xy} bending moment
4. Q_{xx} shear resultant
5. Q_{yy} shear resultant
6. N_{xx} normal resultant
7. N_{yy} normal resultant
8. N_{xy} normal resultant
9. Thickness

Presently, history data for one-dimensional elements, bars and beams, is not processed. Stress data in the ICODE=3 database is not processed.

Modification of TAURUS to accept other databases is a trivial job - at least for the authors.

ACKNOWLEDGEMENTS

The authors would like to thank Bob Bailey for contributing the lateral rail impact example and Appendix B.

Special thanks goes to Nikki Falco who patiently and skillfully typed this report.

REFERENCES

1. J. O. Hallquist, "User's Manuals for DYNA3D and DYNAP," (Nonlinear Dynamic Analysis of Solids in Three Dimensions), University of California, Lawrence Livermore National Laboratory, Rept. UCID-19156 (1981).

2. B. E. Brown, "Displaying the Results of Three-Dimensional Analysis Using GRAPE, Part One - Vector Graphics," University of California, Lawrence Livermore National Laboratory, Rept. UCID-18507, (1979).

3. J. O. Hallquist, "NIKE3D: An Implicit, Finite-Deformation, Finite Element Code for Analyzing the Static and Dynamic Response of Three-Dimensional Solids," University of California, Lawrence Livermore National Laboratory, Rept. UCID-18822 (1981).

4. W. E. Mason, "TACO3D - A Three-Dimensional Finite Element Heat Transfer Code," SAND83-8212, Sandia National Laboratories, Livermore, CA. (1983).

5. R. C. Murray, "User's Manual for GEMINI", University of California, Lawrence Livermore National Laboratory, Rept. (to be published) (1982).

6. J. O. Hallquist, "NIKE2D: An Implicit, Finite-Deformation, Finite-Element Code for Analyzing the Static and Dynamic Response of Two-Dimensional Solids," University of California, Lawrence Livermore National Laboratory, Rept. UCRL-52678 (1979).

7. J. O. Hallquist, "NIKE2D - A Vectorized, Implicit, Finite Deformation, Finite Element Code for Analyzing the Static and Dynamic Response of 2-D Solids," University of California, Lawrence Livermore National Laboratory, Rept. 19677 (1983).

8. J. O. Hallquist, "DYNA2D - An Explicit Finite Element and Finite Difference Code for Axisymmetric and Plane Strain Calculations, (User's Guide)," University of California, Lawrence Livermore National Laboratory, Rept. UCRL-52429 (1978).

9. P. J. Burns, "TACO2D - A Finite Element Heat Transfer Code," (A Modified and Extended Version of the TACO Code by W. E. Mason, Jr.), University of California, Lawrence Livermore National Laboratory, Rept. UCID-17980, Rev. 2 (1982).

10. A. B. Shapiro, "TOPAZ - A Finite Element Heat Conduction Code for Analyzing 2-D Solids," University of California, Lawrence Livermore National Laboratory, Rept. UCID-20045 (1984).

11. J. O. Hallquist, "ORION: An Interactive Post-Processor for the Analysis Codes NIKE2D, DYNA2D, and TACO2D," University of California, Lawrence Livermore National Laboratory, Rept. UCID-19310 (1982).

12 J. O. Hallquist, MAZE - An input generator for DYNA2D and NIKE2D, University of California, Lawrence Livermore National Laboratory, Rept. UCID-19029, Rev. 1 (1982).

13. H. N. Christiansen and M. B. Stephenson, "MOVIE.BYU - A General Purpose Computer Graphics Display System," Proceedings of the Symposium on Applications of Computer Methods in Engineering, University of Southern California, Los Angeles, V. 2, August 1977, pp. 759-769.

14. B. E. Brown, "Structural Analysis Movie Post-Processor (SAMPP) User's Manual," University of California, Lawrence Livermore National Laboratory, Rept. UCID-30097, Rev. 1 (1975).

APPENDIX A

Color Dicomed Output

Briefly, a file ("TSRL" on the 7600 and "TSRL76" on the Crays), is written on the disc whenever the "COLOR" command is activated. This file must be copied to tape for plotting on the Dicomed D-48 color film recorder. The utility routine TAPECOPY found on the CDC7600's should be used as follows (user input in **boldface**):

TAPECOPY 510 60000 (R)
OK.
TSRL76 D. **+BROWN** T. **100 (R)**
OPTIONS OR CARRIAGE-RETURN
(R)
WAITING ON TAPE ASSIGNMENT
ALL TAPES ASSIGNED.

DOUBLE END-OF-FILES READ ON INPUT FILE. CONTINUE?
N **(R)**
η END-OF-FILES READ. TYPE I (NEW INPUT FILE), O (NEW OUTFILE FILE), CARRIAGE-RETURN (CONTINUED), END (TERMINATE), OR OPTIONS **END (R)**
END (R)
ALL DONE

where (R) indicates hitting the return key on the terminal. The file, "TSRL76", must be transported to a 7600 computer without changing its format.

The result is an unclassified tape labeled "BROWN" written with 3*number of frames +1 (η) E.O.F.'s on the tape. This tape should be picked up and taken to the tape rack outside the Graphics Research Lab (Bldg. 113 basement) for plotting. The form shown in Fig. A1 must be filled out as shown and attached to the tape before leaving the tape in the rack of plotting. These forms are available next to the tape rack. On this form "scale" is equal to 4096/resolution. Since TAURUS defaults to 1024, the number 4 is put into the blank space.

REQUEST TO PLOT DICOMED JOB

tape name BROWN date May 15, 1984
submitter Bruce Brown box # S14
dept or div ME classification (UNCL, PARD, etc) Uncl.
phone ext 801-225-0983

job length (approx): (n-1) 3 frames (n-1) minutes -approximate

check camera to use: ___ 35mm X F35mm ___ 16mm ___ 4x5 inch

film to use: X color ___ black and white 4x5 type ________

check PROGRAM to use; check or fill in values for OPTIONS to use:

___ FR80	___ POINT	X SRL
options:	#X, #Y ____	scale 4
____________	scale ____	# pictures (n-1)/3
	# pictures ____	# colors 3
___ DREAD	# colors ____	linear? Y
	filter # ____	compensated? N
___ OTHER ________	invert intensities? ___	double frame? Y
options:	linear? ____	filter seq N (if different)
____________	compensated? ____	rotate? (F35) Y
	double frame? ____	
	6 or 8 bit? ____	
	rotate? (F35) ____	

REMARKS:

Please mount slides.

date & time plotted ____________ # frames ____ time used ____

FIG. A1. Dicomed form filled out. This form is attached to the tape before the tape is left in the rack.

APPENDIX B

Below, several possible colors are defined in terms of their red, green, and blue components. By varying the components, any desired color shade can be obtained.

Color Diomed Plotting

COLOR	RED	GREEN	BLUE
Red	1.0	0.0	0.0
Light red	1.0	0.2	0.2
Yellow	1.0	1.0	0.0
Light yellow	1.0	1.0	0.2
Brown	0.2	0.2	0.0
Green	0.0	1.0	0.0
Light green	0.2	1.0	0.2
Cyan	0.0	1.0	1.0
Light Cyan	0.2	1.0	1.0
Blue	0.0	0.0	1.0
Light blue	0.2	0.2	1.0
Magenta	1.0	0.0	1.0
White	1.0	1.0	1.0
Gray	0.5	0.5	0.5
Black	0.0	0.0	0.0

Log dicomed plotting produced greater depth perception but at far less intensity. Log plotting requires greater intensity blending (i.e. brown is 0.5 0.5 0.0). A diffusivity of 0.35 should be used for log plotting.